UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
PROGRAMA DE PÓS-GRADUAÇÃO EM COMPUTAÇÃO

LUCIANO ZEMBRUZKI

`dnstracker`: Measuring Centralization of DNS Infrastructure in the Wild

Dissertation presented in partial fulfillment
of the requirements for the degree of
Master of Computer Science

Advisor: Prof. Dr. Lisandro Zambenedetti
Granville

Porto Alegre
April 2020

CIP — CATALOGING-IN-PUBLICATION

Zembruzki, Luciano

`dnstracker`: Measuring Centralization of DNS Infrastructure in the Wild / Luciano Zembruzki. – Porto Alegre: PPGC da UFRGS, 2020.

76 f.: il.

Thesis (Master) – Universidade Federal do Rio Grande do Sul. Programa de Pós-Graduação em Computação, Porto Alegre, BR–RS, 2020. Advisor: Lisandro Zambenedetti Granville.

1. DNS. 2. Measurement. 3. Centralization. 4. Colateral Damage. I. Granville, Lisandro Zambenedetti. II. Título.

“All wins conceal an abdication”

— SIMONE DE BEAUVOIR

ACKNOWLEDGEMENT

First of all thank God for the opportunity of life and for giving me health and courage in this study walk. And also for being every day feeding the desire to continue growing.

I thank my parents Orlando and Salete for all their support and dedication and understanding, providing me with the necessary help at all times. My sister for being always present helping me with her enormous goodwill and affection. To my nephew Antony for making my days happier by encouraging me more and more by continuing. I would also like to thank my brother-in-law for his support, who was always ready to help when needed.

To the teachers for all the teachings given to me in all these college years. In particular to my advisor Doctor Lisandro Zambenedetti Granvile for all the orientations, teachings and experiences shared with me, without which this work would not have been accomplished.

Thank the INF colleagues who always encouraged me to keep improving.

Thank my friends who were or are part of my life. By understanding the abdication of some moments with them to accomplish the tasks of the master's.

ABSTRACT

The Internet Domain Naming System (DNS) is one of the pillars of the Internet and has been object of a number of Distributed Denial-of-Service (DDoS) attacks over the years. As a countermeasure, DNS infrastructure has been programmed to include a series of replication measures, such as relying on multiple authoritative DNS servers and the use of IP anycast. Even though these countermeasures have been in place, it has been found that, when servers rely on third-party DNS providers for reliable services, there may be a certain degree of infrastructure centralization. In this case, an attack against a DNS target might affect other authoritative DNS servers that share a part of the infrastructure with the intended victim. However, measuring these kinds of infrastructure sharing is a daunting task, given that generally researchers do not have access to internal DNS provider. In this work, an attempt is made to set out a solution that is supported by a `dnstracker` tool that uses active DNS measurements to determine, the varying levels of shared infrastructure. As a case study, we analyze the authoritative name servers of all the domains of the most visited websites in the Alexa Top 1 Million List. Our results show that, in some cases, up to 12,000 authoritative name servers share the same underlying infrastructure of a third-party DNS provider. This means that, in the event of an attack, these authoritative DNS servers have increased their risk of suffering from collateral damage.
Keywords: DNS. Measurement. Centralization. Colateral Damage.

`dnstracker`: Medindo a centralização da infraestrutura DNS na Internet

RESUMO

O Sistema de Nomes de Domínio (Domain Name System - DNS) é um dos pilares da Internet e foi alvo de vários ataques DDoS (Distributed Denial-Service - Denial of Service) ao longo dos anos. Como uma medida contrária, a infraestrutura DNS foi projetada com uma série de técnicas de replicação, como confiar em vários servidores de nomes com autoridade e usar o IP anycast. Embora essas medidas estejam em vigor, vimos que, quando os servidores contam com provedores de DNS de terceiros para serviços autorizados, pode haver certos níveis de centralização da infraestrutura. Nesse caso, um ataque contra um destino DNS pode afetar outros servidores DNS autorizados que compartilham parte da infraestrutura com a vítima pretendida. No entanto, medir esses níveis de compartilhamento de infraestrutura é uma tarefa desafiadora, uma vez que os pesquisadores normalmente não têm acesso aos internos do provedor de DNS. Nesta dissertação, apresentamos uma metodologia e a ferramenta `dnstracker` associada, que permitem medir, em vários graus, o nível de concentração e infraestrutura compartilhada usando medidas de DNS ativas. Como estudo de caso, analisamos os servidores de nomes com autoridade de todos os domínios dos sites mais visitados do Alexa Top 1 milhão. Nossos resultados mostram que, em alguns casos, até 12,000 servidores de nomes autorizados compartilham a mesma infraestrutura subjacente de um provedor DNS de terceiros. Como tal, no caso de um ataque, esses servidores DNS autorizados aumentaram a probabilidade de sofrer danos colaterais.

Palavras-chave: DNS. Medições. Centralização. Dano Colateral.

LIST OF ABBREVIATIONS AND ACRONYMS

ASN	Autonomous System Number
AS	Autonomous System
BGP	Border Gateway Protocol
CSV	Comma Separated Values
DDoS	Distributed Denial-of-Service
DIG	Domain Information Groper
DNS	Domain Name System
FQDN	Fully Qualified Domain Name
HBTL	Hop Before the Last
HTTP	Hypertext Transfer Protocol
ICMP	Internet Control Message Protocol
IP	Internet Protocol
ISP	Internet Service Provider
IoT	Internet of Things
JDK	Java Development Kit
JNI	Java Native Interface
JSON	JavaScript Object Notation
MVC	Model View Controller
ORM	Object-Relational Mapping
PID	Process Identifier
REST	Representational State Transfer
RFC	Request for Comments
SLD	Second-level Domain
TLD	Top-level Domain

TTL	Time to Live
URI	Uniform Resource Identifier

LIST OF FIGURES

LIST OF TABLES

CONTENTS

1 INTRODUCTION

The Internet Domain Naming System (DNS) provides a globally hierarchical naming space on the Internet that enables mapping of hosts, networks, and services to IP addresses (MOCKAPETRIS, 1987a; MOCKAPETRIS, 1987b). Thus, along with other services, DNS is one of the core services of the Internet. When seeking to resolve a domain name (*e.g.*, `www.google.com`), a client first sends a DNS query to its *DNS recursive resolver* ("resolver" hereafter), which is a DNS server that can resolve the domain name on behalf of the client. If the resolver does not have a DNS record in a cache, it will query the DNS hierarchy for a response. Resolvers are responsible for sending queries to *authoritative DNS nameservers*, which are the servers responsible for providing answers to resolvers about the fetched domain. These authoritative DNS servers are divided into zones and know the content of a DNS zone on the basis of local knowledge, and thus can answer queries about these zones (ELZ et al., 1997). For example, a client could connect to `1.1.1.1` (1.1.1.1, 2018) – a public DNS resolver – asking for the IP of `www.google.com`. The resolver will, send queries to the authoritative DNS servers of `www.google.com` on behalf of the user, which are `ns1.google.com` and `ns2.google.com`, and return the desired IP address to the client.

1.1 Problem and Motivation

Since they represent one of the essential services for the Internet, authoritative DNS servers have often been victims of Distributed Denial-of-Service (DDoS) attacks. These DNS servers, have been targeted on various occasions in the last decade (VIXIE; SNEERINGER; SCHLEIFER, 2002; SENGUPTA, 2012; OPERATORS, 2015; OPERATORS, 2016; WEINBERG M., 2016; MOURA et al., 2016), and even DNS providers have been victims of attacks (HILTON, 2016), which have disrupted many of their domains (PERLROTH, 2016). To curb these kinds of attacks, *layers of replication* have been designed in the DNS:

- A domain name may use multiple authoritative DNS servers (MOCKAPETRIS, 1987a);
- Each authoritative DNS server may employ IP anycast (MCPHERSON et al., 2014), which allows the same IP addresses to be replicated and announced at various loca-

tions, referred to as “anycast sites”;

- Each site, in turn, may use local load balancers to distribute queries among multiple authoritative DNS servers;
- DNS Nameservers are required to have geographical and topological diversity (ELZ et al., 1997).

Even though these measures are broadly employed, when domain names share the same DNS provider, they may (whether unknowingly or not) be sharing different levels of *infrastructure*, such as pipes, servers, and data centers (ABHISHTA; RIJSWIJK-DEIJ; NIEUWENHUIS, 2019). As many companies do not run their own DNS infrastructure, but rely on outsourcing to third-party DNS providers instead, detecting the extent of infrastructure sharing among many of the different domains hosted by these DNS providers, is a challenging task. Along with this outsourcing trend, many companies have also started to broadcast their IP blocks, which originate from datacenters hosted by third-party DNS providers. This trend might be leading to the centralization of the DNS ecosystem and may become a problem if a significant DDoS attack takes place. Moreover, if parts of the shared infrastructure become overwhelmed, all the DNS zones in the service may experience problems too. As a result, many domains in certain zones can become unreachable. The Dyn attack (HILTON, 2016; NEWMAN, 2016) exemplifies the *collateral damage* which can be caused when authoritative DNS servers hosting multiple DNS zones, are under attack. In the case of this particular attack, the Dyn servers were unable to process the users’ DNS requests, and as a result, the users could not obtain access to web domains that had contract with Dyn, such as Twitter, GitHub, and Netflix. The fact that the DNS infrastructure comprises many servers, and these are traditionally distributed in a wide range of locations, makes it more resistant to unexpected errors. However, despite this, it remains vulnerable to targeted attacks against infrastructural components that have become centralized either by practice or design.

The DNS ecosystem has been analyzed and studied by a number of authors (AGER et al., 2010; PERDISCI; CORONA; GIACINTO, 2012; BILGE et al., 2014; MUGALI et al., 2015), but few studies have been concerned with measuring different layers of the shared DNS infrastructure. In addition, it should be noted that, measuring these levels of infrastructure sharing is a daunting task, since most researchers do not have access to Internal DNS providers. In light of this, researchers have to resort to active measurements that allow them to estimate, at the IP level, a certain degree of shared infrastructure, or analyze traditional DNS datasets. This has been carry out previously by several au-

thors (MOURA et al., 2016; ALLMAN, 2018; BATES et al., 2018) who analyzed different aspects of the DNS ecosystem, such as the robustness and centralization of Top-Level Domains (TLD) (BATES et al., 2018) (ALLMAN, 2018) and root servers (MOURA et al., 2016). They also succeeded in shedding some light on infrastructure sharing at the TLD level, by providing evidence that network-level infrastructure sharing is becoming increasingly widespread. However, the studies cited above fail to examine the question of DNS centralization in terms of an Autonomous System (AS), but obtain their information by relying solely on third-party DNS providers. As well as this, the authors did not conduct an analysis of the shared infrastructure of authoritative DNS servers for Fully Qualified Domain Names (FQDN).

1.2 Aims and Main Contributions

Against this background, in this dissertation we aim to introduce a flexible solution that assesses the degree of centralization of authoritative DNS servers by means of active DNS measurements. Our study focuses on analyzing a possible concentration of authoritative DNS servers in the wild, for FQDNs. In addition, we designed `dnstracker`, an opensource tool that implements our proposed solution and assists in consolidating our findings. As a case study, we use `dnstracker` to analyze all the domains of Alexa Top 1 Million List websites (ALEXA, 2018c).

The main contributions achived by this work are as follows:

- The design of an active measurement solution to evaluate the level of centralization of the DNS infrastructure for FQDNs;
- An open-source tool `dnstracker` that allows the solution to be employed;
- A large-scale measurement of the DNS infrastructure;
- A detailed analysis of the impact that the centralizing DNS infrastructure has on the Internet from the datasets created by the design tool;

This study found that, in some cases, up to 12,000 authoritative DNS servers of the most visited websites share the same infrastructure as the DNS providers, and for this reason, risked suffering from collateral damage in the event of an attack. This means that, in the event of a successful attack on this AS, over 77,419 websites would be unreachable, since the clients would not be able to resolve the Fully Qualified Domain Names - FQDNs.

1.3 Outline of the Research Study

This dissertation is structured as follows. Background and Related work are discussed in Chapter 2,where there is a review of the concepts of the DNS infrastructure. In addition, in this chapter, we review studies in the literature that analyzed DNS and its infrastructure. In Chapter 3, there is an outline of the solution used to measure the DNS centralization and its effectiveness. We then introduce `dnstracker` and how it can be implemented in Chapter 4. There is a discussion of our results in Chapter 5. Finally, in Chapter 6, we conclude this work with some final remarks, along with recommendations for future work.

2 BACKGROUND AND RELATED WORK

This chapter provides an overview about different components that make up the DNS and explains how their features may be representative. In Section 2.1 below, we begin with a brief discussion about the origins of the DNS. In Section 2.2, we define some concepts and terms about Domain Names. In Section 2.3, we set out the DNS infrastructure. In Section 2.4, we describe the process of a domain name resolution. In Section 2.5, we discuss the related works. In the first part of this section there is an examination of some works that use DNS measurements. In the second part, there are some papers that focus on centralizing the DNS infrastructure. In Section 2.6, we define the problem statement based on an investigation of how an attack can interfere with the name resolution process and how harmful it can be.

2.1 The Domain Name System - DNS

The practice of naming hosts on the network has been in use almost since the dawn of the Internet. Initially, all the sites connected to the home network had a copy of a file called the *HOSTS.TXT*. This file provided the mapping of names for network addresses (STEWART, 2019). However, it was realized that keeping separate copies of this file synchronized to a growing network was impratical. By early 1974, there were still fewer than fifty hosts on ARPANET (STEWART, 2019). When the first Request for Comments - RFC document on host naming was written (DEUTSCH, 1973). This document, and those that followed, specified how and where the list of host-to-address mappings should be hosted. However, maintaining a central database was prone to errors. In early 1982, problems with relaying mail over the network led to the beginning of the current concept of hierarchical domain names in a structured manner (POSTEL, 1982). The first Domain Name System specification set appeared in 1983 (MOCKAPETRIS, 1983a; MOCKAPETRIS, 1983b) In 1987, the DNS specifications were updated, and this resulted in the basic protocol that still remains in use today (MOCKAPETRIS, 1987a; MOCKAPETRIS, 1987b).

2.2 Concepts and Terms of Domain Names

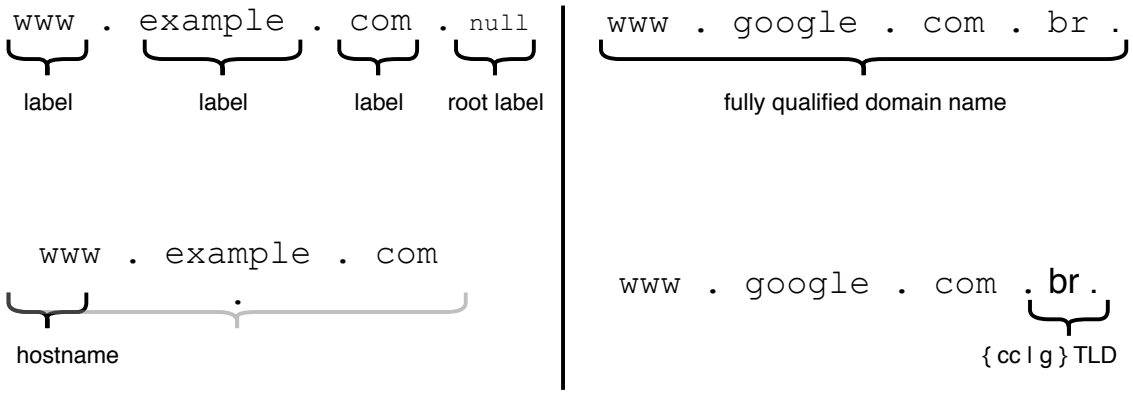
A structured ASCII string represents a domain name. This system means that, domain names are created from dot-separated labels. Figure 2.1 shows examples of domain names with concepts that are used in this study. The left-hand side of the figure 2.1 presents the following terms related to domain names:

Label: Domain names are made up of labels. Labels are not case sensitive, (*i.e.,*"www" and "WWW") are equivalent. In DNS message exchanges, each label is encoded using a single unsigned byte value that indicates the size of the label, followed by 8-bit ASCII characters for the labelled text.

Root Label: The root label is the end of a domain name and is represented as a null label. The root label is indicated to by a single dot at the end of the name, but this dot is usually omitted. In DNS message exchanges, the root label is represented as a single byte value set to `0x00`. This label indicates the top of the DNS hierarchy (which will be discussed in Section 2.3).

Hostname: In some cases, this term refers to the left-most label of a domain name, in which case it usually refers to the local name of a machine. In other situations, the term may be used to refer to an entire domain name. On account of this ambiguity, we try to avoid using this term in this dissertation.

Figure 2.1: Domain name concepts



Source: Adapted from (RIJSWIJK-DEIJ, 2017)

The right-hand side of the figure shows the following terms:

Fully Qualified Domain Name: Abbreviated to FQDN, the term means the entire domain name, that is, all the labels that make up the name, including the root label. In this dissertation, the term "domain name" refer to an FQDN.

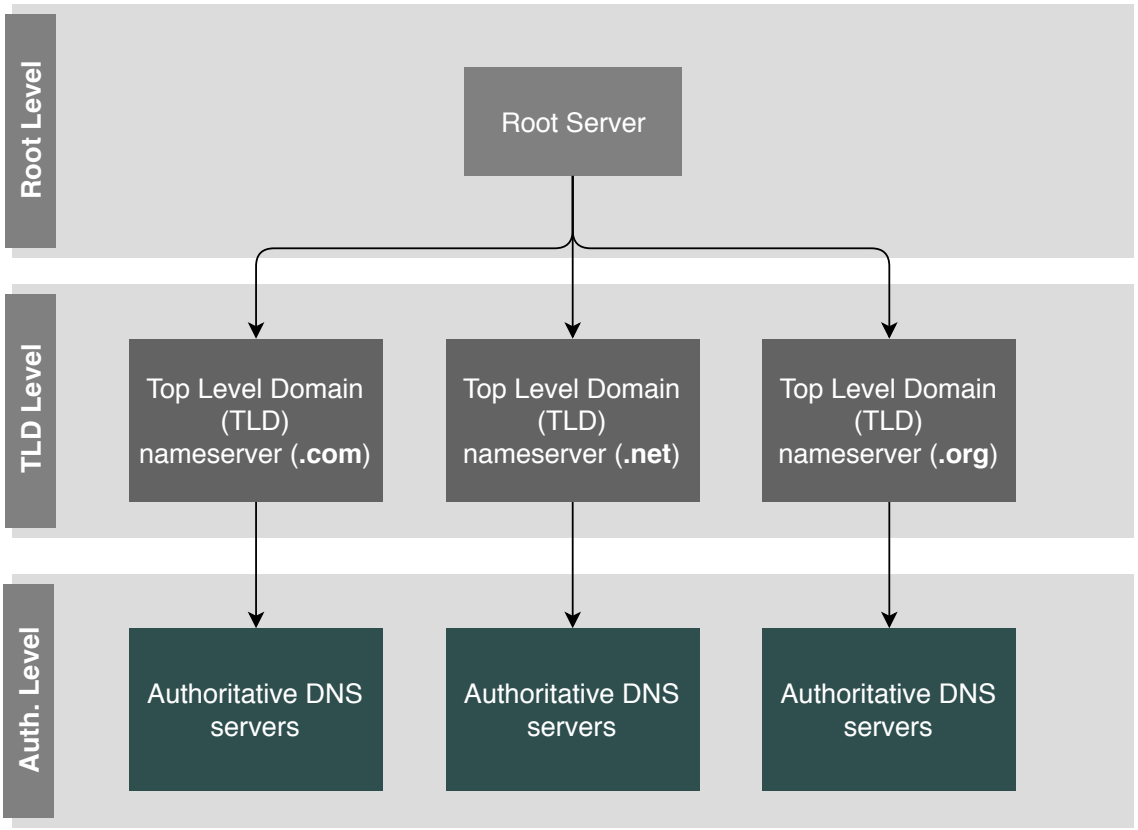
{cc | g }TLD: TLD is short for Top Level Domain. TLDs are domain names

directly below the root in the DNS hierarchy (discussed in the 2.3 section). The terms ccTLD and gTLD are also used. The first, “cc,” indicates the country code, as these TLDs specifically refer to geographical countries. The last, “g,” refers to Generic. Generic TLDs are not country-specific

2.3 DNS Hierarchy

The Domain Name System (DNS) is structured as a hierarchically distributed database (CHANDRAMOULI; ROSE, 2006). When seeking to access Internet resources by means of user-friendly domain names rather than IP addresses, users need a system that maps the domain names to IP addresses. This translation is the primary task of an engine called the Domain Name System (DNS). The DNS infrastructure comprises of geographically distributed computing and communication entities worldwide. It is necessary to examine the structure behind the organization of domain names first to understand the DNS structure (LIU; ALBITZ, 2006).

Figure 2.2: Example of the DNS Hierarchy



Source: Adapted from (BATES et al., 2018)

Figure 2.2 shows that the DNS is arranged in the form of a hierarchy.

- **Root DNS Servers** - At the top of the hierarchy are the "Root" servers. These Root servers are responsible for storing the records corresponding to the next level in the hierarchy, which is the top-level domain name (TLD) servers.
- **Top-level Domains - TLDs** - TLDs (including `.com`, `.net`, `.org` and country-level identifiers such as `.nl` or `.br`) can be found at the far right of the URL addresses used every day. Each TLD nameserver is responsible for keeping the records corresponding to the authoritative DNS servers of the domains that fall under that TLD. For example, `.com` DNS servers maintain records for the domain namespace of the authoritative DNS servers, and cover domain names such as `www.google.com` and `www.netflix.com`.
- **Authoritative DNS servers** - The Authoritative DNS servers, which are the last, can be hosted by one's infrastructure or by third-party providers, like Dyn or NetNod. These authoritative DNS servers are responsible for mapping individual domain names into IP addresses.

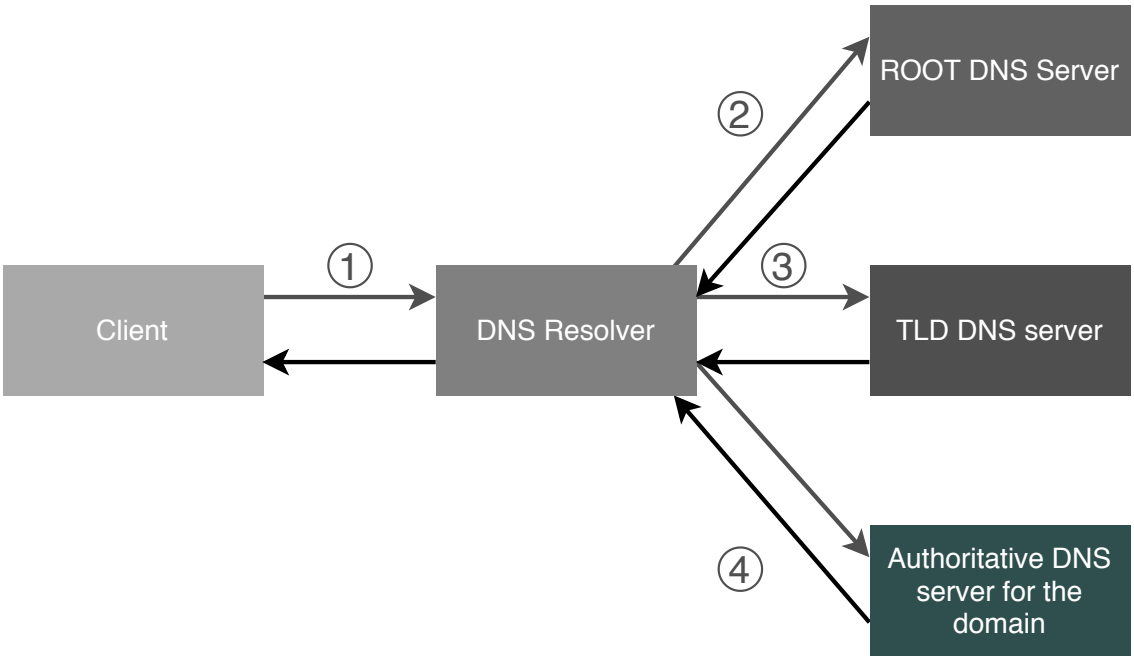
2.4 DNS Resolution

In this section, we describe how the process of resolving a domain name works. Let us trace the steps of a client who wants to obtain a page or service of an individual domain name. It must use the DNS system to get the IP address.

If a client wants to resolve a domain *e.g.* `www.google.com`, it is first necessary to submit a DNS request to a *resolver*, which is a DNS server that can resolve the domain name on behalf of the client. Such resolver tracks the labels of this domain separated by a dot (.) from right to left to search for the required authoritative DNS server. Figure 2.3 shows the process of resolving a domain name *e.g.* `www.google.com` to ilustrate how this process works.

1. **Query to a resolver** - In the first stage, (1), the client sends a query to its resolver requesting the domain (`www.google.com`). This resolver is often the client's Internet Service Provider (ISP).
2. **Query to an authoritative name server for the root** - Next, in Stage (2), the resolver will start sending the Query to one of the authoritative Root servers asking for `google.com`. The Root name servers do not have authority to act on behalf

Figure 2.3: The process of resolving a domain name



Source: Adapted from (CISCO, 2017)

of `google.com`, so they cannot answer the Query. However, the root server delegates this task to `.com` servers, so the root name server will respond to the list of authoritative name servers for the `.com` TLD.

3. **Query to `.com` authoritative DNS server** - In Stage ③, the resolver queries the `.com` TLD DNS server with the second label of the domain name ("`google`"). Again, these servers have no authority for `google.com` and do not know the answer. Figure 2.4 exemplifies the list of authoritative name servers for `google.com` delegated by `.com`.

Figure 2.4: Delegation for `google.com` in the `.com` zone

domain name	TTL	class	**type**	value
google.com.	172800	IN	NS	ns1.google.com.
google.com.	172800	IN	NS	ns2.google.com.
google.com.	172800	IN	NS	ns3.google.com.
google.com.	172800	IN	NS	ns4.google.com.

Source: The Author

4. **Query to `google.com` authoritative DNS server** - In Stage ④, the resolver finally sends a query to `www.google.com` or to one of the `google.com` name servers. Since they have the authority to act for the domain being looked for, they

will return the IP address linked to `www.google.com` in the requested response.

5. **Repling to the client** - Finally, in Stage (5), the resolver replies to the client with the appropriate IP address (CISCO, 2017).

2.5 Related Work

In this section, we discuss previous attempts to make DNS measurements. Some researchers took measurements related to the robustness of the DNS infrastructure in the past, by analyzing different factors related to possible points of failure in the DNS ecosystem. In particular, it is worth highlighting three past studies (MOURA et al., 2016; BATES et al., 2018; ALLMAN, 2018) that provide evidence of a shared DNS infrastructure and underline the dangers that arise from it. Some works are discussed below that are related to DNS measurements in general. These studies are important because they show some of the difficulties faced when conducting studies and making measurements within the DNS ecosystem.

In (FOMENKOV et al., 2001), the authors described features of active end-to-end latency and topology measurements among various DNS Root servers and a set of their clients, by using the CAIDA skitter tool. The objective of this work was to create an analytical framework to evaluate the optimization of the root server location and the effect it had on the efficiency of DNS services. The authors gathered together a sample of clients for each monitored DNS root server, divided these samples into a list of common destinations, and then actively analyzed these destinations and their connectivity. The destination subsets that have significant latency connections to all the root name servers were identified, and their geographical location was discussed.

According to (LIU et al., 2007), DNS root name servers routinely use anycast to improve customer service and increase resilience against various types of failure. The authors estimated the amount of DNS traffic collected over two days in January 2006 on anycast instances to root nameservers C, F, and K. It has been observed that anycast's DNS service affects Internet users throughout the world. When determining if clients use their closest instance, we searched the locations of clients for each Root instance and the geographical distances connecting a server and its customers. It has often been found that the choice, which is solely determined by BGP routing, is not the closest geographically. The authors also considered specific AS paths and investigated some cases where local instances have a higher than usual proportion of non-local clients. Overall, the work has

shown that anycast's roots significantly assist in locating DNS traffic, and improves the DNS service for customers worldwide.

Studying the infrastructure of large parts of DNS over time, reveals valuable information about how the Internet has evolved. Compiling a long-term dataset with daily DNS measurements, requires a good deal of effort and making measurements that have to address the challenge of carrying out active metering on a large scale (*e.g.*all top-level domains (TLDs)) on the Internet (`.com`, `.net` and `.org`) which corresponds to 50% of the global DNS namespace). The study by (RIJSWIJK-DEIJ et al., 2016) discusses the design options that have been selected to address these challenges and document the designing of the measurement system. The data from these collections are significant to the network research community. In view of this, it is essential to discuss how to make data accessible to other researchers.

The Domain Name System contains a great deal of information about Internet security, stability, and health. Most searches that use DNS to detect malicious activity do so by making use of passive measurements. This approach has limitations however, since it is only valid when an attack is in progress. In (SPEROTTO; TOORN; RIJSWIJK-DEIJ, 2017), the authors argue in favor of using active DNS measurements for the proactive identification of maliciously configured domains. The survey uses data from OpenINTEL: a large-scale active DNS measurement project, which since February 2015, has been collecting daily snapshots of over 60% of the current DNS namespace. The authors show their preliminary results in three case studies, namely snowshoe spam, denial of service attacks, and a targeted phishing case known as CEO fraud.

Moura *et al.* (MOURA et al., 2016) analyzed the DDoS event suffered by the DNS root servers in 2015. From Nov. 30th to Dec. 1st, 2015, many of the Root DNS Letter Servers had an unusually high rate of specific requests, with a traffic load a hundred times larger than normal. The authors stated that, even though these episodes did not target specific end-services, there was evidence of Internet services suffering from collateral damage because they shared their DNS provider infrastructure with the DDoS target. In the 2015 attack, some `.nl` TLD servers were taken down as a side effect of the attack on the DNS Root server. However, even though the subsequent investigation diagnosed the cause of the events and provided some evidence of centralization as a side effect, it failed to provide an in-depth examination the possible level of centralization in the DNS infrastructure.

Bates *et al.* (BATES et al., 2018) put forward a solution to measure how far the

global DNS has retained its distributed resilience, given the rise of cloud-based hosting and a new infrastructure. In their work, the authors analyzed the trends for a greater concentration and diversification of the DNS infrastructure over a period of time, when they sampled the 1,000 main US domains in the TLDs `.com`, `.net`, and `.org` in accordance with Alexa Top Sites (ALEXA, 2018c). The authors also pointed out that their analysis concentrated on the traditional domains (`.com`, `.net`, and `.org`) because they are among the oldest TLDs, and thus represent a broad spectrum of the current Internet. However, the authors recognize that their results might change if other regions, such as `.ru` and `.cn`, are taken into account. However, although it provided an insight into the robustness of DNS, the work did not take note of the authoritative DNS server, which is a crucial factor in the reliability of the infrastructure. This, in turn, is covered by our work.

Allman *et al.* (ALLMAN, 2018) carried out a study to determine the robustness of the DNS ecosystem and their analysis focused on second-level domains (SLDs) (*e.g.*, `icir.org`). In this study, the authors used two sets of zone data for the `.com`, `.net`, and `.org` TLDs, which were collected over a period of nine years. They also conducted an analysis of DNS infrastructure sharing. Initially, it was noted that in the dataset, 91% to 93% of the SLDs share at least one name server (by IP) and even worse another SLD. In an approach based on individual SLDs, it has been noted that half of the SLDs share exactly one set of name servers with at the very least 163 other SLDs. In addition, it was discovered that the largest group contains 9,000 SLDs that share the same set of name servers. In the next stage, there was a network-based sharing analysis (IP blocks). In this search, it was found that the infrastructure was more widely shared when viewed from a network perspective than from a host perspective. In addition, the authors point out that infrastructure sharing at network-level is becoming increasingly more common. Finally, they analyzed the data to determine whether there is more often a shared infrastructure in domains with a higher or lower ranking.

In light of the research carried out so far by the scientific community, there is strong evidence that suggests some level of DNS centralization. However, these studies have had a limited scope, as the coverage of DNS records and the number of domains that need to be measured is constantly increasing, and none of the works in the state-of-the-art has explored the extent to which the authoritative DNS servers for Fully Qualified Domain Names - FQDNs have been centrilized. Furthermore, it should bear in mind not only the *Last Hop* but the *Hop-Before-the-Last* too, which is a part of the work undertaker here. Thus, measuring DNS comprehensively, on a large scale and for the long term,

remains a significant challenge. However, this kind of measurement can provide valuable information about how the Internet has evolved, and that is what we plan to do. We seek to undertake a comprehensive measurement of the DNS infrastructure to understand the concept of centralization and its risks, on the basis of the collected data. Another key factor in our work is the provision of an online tool which is designed to detect if it is probable that a service will be directly affected by other domains that share the same DNS infrastructure.

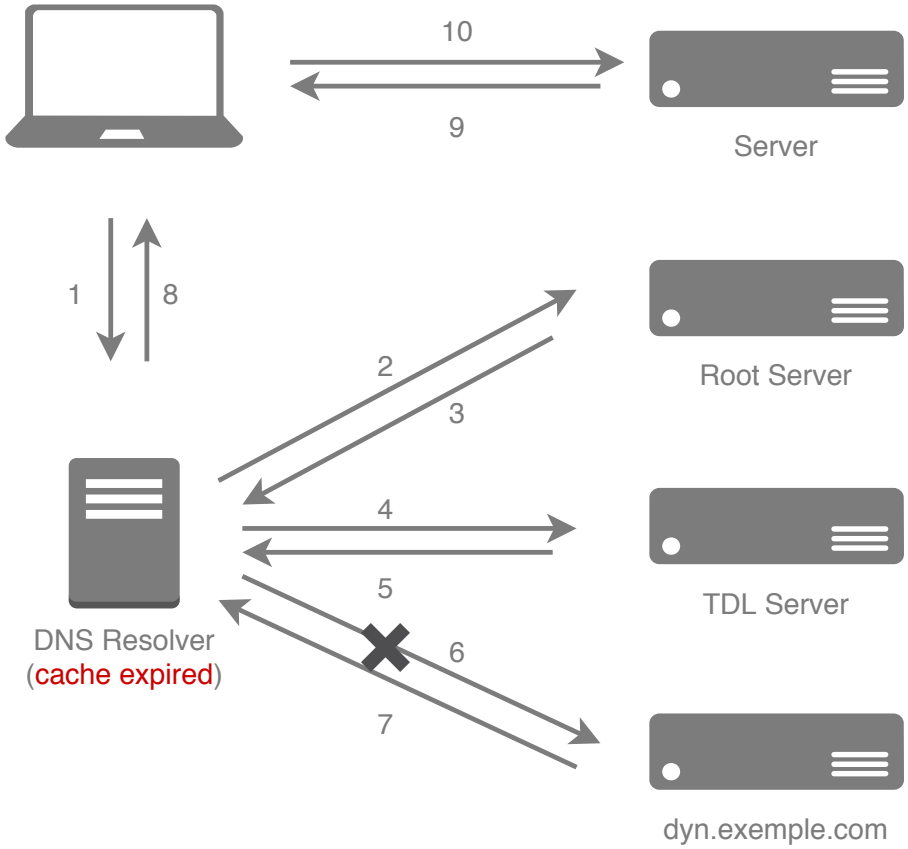
2.6 Problem to resolve a domain

In this section, we examine the problem statement. Collateral damage cannot be precisely assessed by a simple analysis of the IP addresses of different authoritative DNS servers. However, different servers – each operating under their IP address block – can be hosted within the same service providers' infrastructure. Moreover, owing to the commercial nature of the DNS providers, the data required to analyze this kind of aggregation is rarely disclosed. The problem can, however, be evaluated if a common node (or single point of failure) can be found in the routing path to a particular set of remote servers. For instance, if we obtain the IP address of `a.dns.nl`, one of the authoritative DNS servers for the `.nl` zone, and examine its AS, we will find it belongs to a DNS provider. In fact, these authoritative DNS server are run by NetNod in Sweden. Hence, if other authoritative DNS servers are hosted in the same Netnod infrastructure, they will agree to share the collateral damage in the event of a potential DDoS attack. We discuss below how an attack can interfere with the name resolution process and how harmful it can be. The attack on Dyn has been selected as being wholly representative of the problem in question. It should be emphasized that attacks like this have become more frequent and serious (SAHARAN; GUPTA, 2019) and have similar causes and effects. In this attack on Dyn, access to many domains depended on the resilience and stability of a single DNS provider (FILIPPI; MCCARTHY, 2012).

Fortunately, records are cached at various points in the process to resolve a domain. However, these cached records can become inaccurate over time and eventually expire. Most records of larger sites have much shorter expiration periods (LIU; ALBITZ, 2006; BATES et al., 2018). If the authoritative DNS servers for a specific domain name go down (as many of those administered by Dyn did in the October DDoS attack) (HILTON, 2016; NEWMAN, 2016), the DNS resolvers will not be able to update the ex-

pired or changed records. This means that a DNS malfunction effectively prevents users from accessing the requested content, even if that content is hosted on a healthy server. Figure 2.5 shows how, because records eventually expire, the resolver, root servers, and TLD nameservers have to rely on the authoritative DNS servers to update their records. When Dyn stopped responding to requests, the expired records could no be updated, and as a result, the resolver could not respond to the DNS requests. As can be seen in Figure 2.5, in the case of Dyn attack, the resolver could not proceed with Stage 6 and hence, the next stages in the domain resolution.

Figure 2.5: Damage caused by the Dyn attack



Source: Author

The Dyn attack provides an illustrative example of how DNS infrastructure vulnerability and a possible centralization of the DNS infrastructure can cause collateral damage to ISP clients sharing their infrastructure. This attack has caused numerous websites to crash, and rendered them inaccessible for many hours, (NEWMAN, 2016; ATTACK, 2018; HILTON, 2016). In addition, this unfortunate attack highlights the problems caused by centralizing a DNS infrastructure since it may lead to the interruption of essential services.

As shown in the hierarchical architecture of Figure 2.2, the DNS infrastructure is distributed by design. (MOCKAPETRIS; DUNLAP, 1988). DNS queries are managed by thousands of different servers around the world rather than by a master server that maps domain names to IPs. It is assumed that distribution provides a degree of redundancy and reliability to the DNS system, by reducing possible "single points of failure," which can cause Internet access problems in case of failure (KUROSE, 2005).

However, the advent of virtualization and use of cloud to host and manage domains can break this distributed architecture. Companies like Cloudflare, Amazon, and Dyn often offer reliable DNS hosting and management services that make life easier for the network operator, and other cloud services are offered too. DNS hosting by third party providers offer advantages in terms of geographic reach, reliability, and load balancing. However, these benefits, can lead to the centralization of DNS services in a small number of providers and this can pose a threat to Internet security in terms of stability and trust. As well as this centralization can lead to "single points of failure," which in turn poses the risk of a massive service downtime.

However, how important is DNS downtime? (KIM et al., 2011) point out that the DDoS attacks came to the public attention in February 2000, when commercial websites were unable to service their customers for hours resulting in damage assessed as worth US$1.7 billion. One Research study (ARMIN et al., 2015) estimated that an average DDoS attack costs the victim company US$40,000 per hour of disruption. The same study found that most attacks last between 6 and 24 hours.

3 DNSTRACKER

In this chapter, there is a description of `dnstracker`. Our planned solution allows the level of centralization of authoritative DNS servers to be assessed, at various levels, with the aid of active DNS measurements. We concentrate on collecting and analyzing this possible centralization within the DNS infrastructure. This involves seeking a solution that makes it possible to collect routing-path information from a vantage point to an authoritative DNS server. Section 3.1 describe the modules and procedures followed in our solution. In Section 3.2 we describe how our solution was designed to work from different vantage points around the world. Finally, in Section 3.3 we describe the data collection and how this data can be used to infer the degree the centralization that can be found in the DNS infrastructure.

3.1 Solution

In this section we define our conceptually-based `dnstracker`. Figure 3.1 shows the modules and procedures followed for the collection of information from the authoritative DNS servers that will make it possible to analyze the centralization of the DNS infrastructure. In Figure 3.1, the three main modules are shown that make up the solution. Each module and its processes are described below:

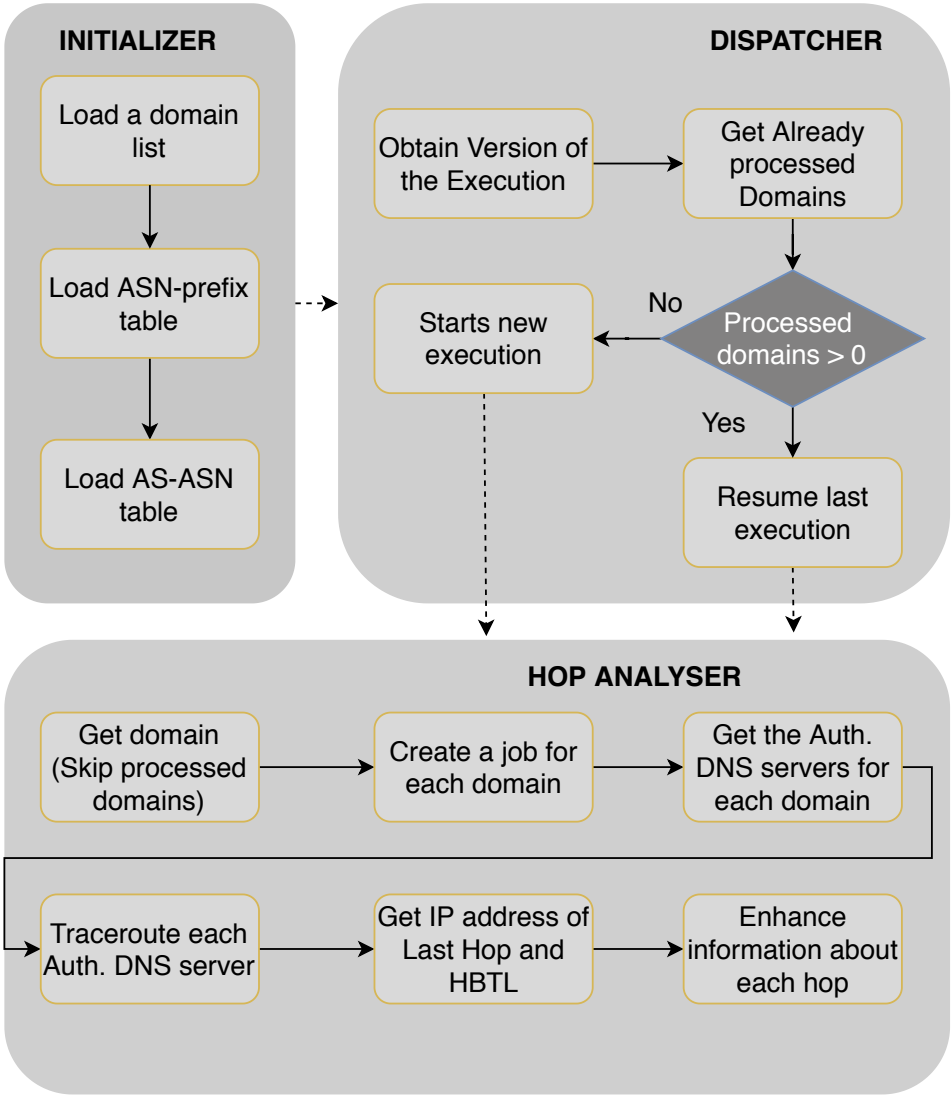
Module (1) - Initializer: This module is responsible for loading the files with the data used by our solution. Each stage of the process performed for this module is described below.

- *Load Alexa Top 1M List*: In the first stage of the procedure the system is loaded with a list of domain names. This entails using a predetermined list of *Fully Qualified Domain Names* - FQDNs. As we wished to obtain a global coverage of domains, we decide to use the Alexa Global Top 1 Milion (ALEXA, 2018c) ranking list. Alexa Global Top 1 Milion is the most popular and most widely used domain list (ALEXA, 2018c). This list is compiled on the basis of web activity monitored by the plugin Alexa browser and includes 25,000 different browser extensions (ALEXA, 2018a; ALEXA, 2018b; ALEXA, 2018d). These are often offered for sale and the Alexa lists have a few free offers as well. The Global Top 1M list is the most popular free offer available but was withdrawn at the end of 2016. Al-

though it was discontinued in 2016, a study of the Internet Top Lists (SCHEITLE et al., 2018) highlighted Alexa's list as being an overall representative example of typical DNS traffic, and as forming a strong basis for an analysis of DNS. In addition, this list continues to be used in research studies published in the area which has the most significant events involving measurements.

- *Load AS-prefix tables*: After this, the tables referring to the mapping of ASN-prefix and AS-ASN are loaded into the system. Publicly available prefix-to-ASN and AS-to-ASN tables (FRANK, 2018) are used to obtain the corresponding AS of the hop, as well as the owner of the hop (*i.e.,* to determine which company is responsible for running the infrastructure). The use of these tables in our solution is described Module *Hop Analyser*.

Figure 3.1: Solution process



Source: The Author

Module (2) - Dispatcher: This module basically controls the versions of collection runs required for our solution.

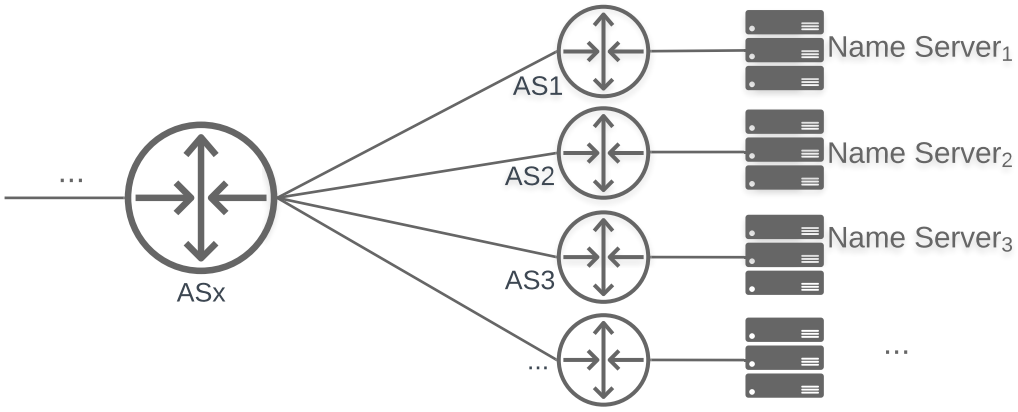
- *Obtain Version of the Execution*: The first stage of this module is responsible for obtaining the identifier of the current execution version.
- *Get Already Processed Domains*: This version of the execution identifier make it possible to find out if it has already processed the domains. It also enables us to check whether this is a new execution or the resumption of a previously unfinished execution.
- *Starts or Resume a execution*: If the number of processed domains is greater than zero ($processedDomains > 0$) it means that this is an unfinished execution. If the number of processed domains is equal to zero ($processedDomains = 0$) it means that this is a new execution. Both process flows call the *Hop Analyzer* module to continue the procedure of our solution.

Module (3) - Hop Analyser: This module is the core of our solution and implements the actions required to collect information from the DNS servers.

- *Get domain (Skip processed domains)*: In the first stage of this module, the domain that has to be processed is obtained from an Alexa Top 1M domain name list. If this execution is a resumption of a previous run, the domains that have already been processed are ignored.
- *Create a job for each domain*: The next stage is to create a job to each domain. This job is being responsible for collecting the DNS information related to this domain.
- *Get the Auth. DNS servers for each domain*: In the next stage, each created job discloses the authoritative DNS servers of the domain. An FQDN usually has two to four separate authoritative DNS servers, but many domains share the same authoritative DNS server.
- *Traceroute each Auth. DNS server*: Following this, we execute a `traceroute` from a given vantage point for every authoratative DNS server. The `traceroute` provides information about the addresses of each hop in the route from the client to the domain's authoritative DNS server. Whenever a different set of servers, owned by different websites, are hosted within the same infrastructure of the provider, requests to these servers will share a common point – this is the network hop just before it reaches its final destination, referred to as *Hop-Before-The-Last* (HBTL). If two different requests are served through a path where the HBTL is in the same

AS, they are likely to be hosted hosted by the same DNS provider, thus sharing the same infrastructure to some extent, as illustrated in Figure 3.2.

Figure 3.2: Name Servers whit a HBTL that shares the same AS infrastructure



Source: The Author

- *Get IP addresses from Last Hop and HBTL*: The IPv4 address in the path to the authoritative DNS server is extracted from each hop obtained through *traceroute*, we extract the IPv4 address of each hop in the path to the authoritative DNS server. However, in our approach, we only store the relevant data of the last hop and the HBTL, as these are the most likely points of infrastructure aggregation in the DNS ecosystem.
- *Enhance information about each hop*: We extract the IPv4 address of each hop in the path to the authoritative DNS server from each hop obtained through `traceroute`. However, in our approach, we only store the relevant data of the last hop and the HBTL, as these are the most likely points of infrastructure aggregation in the DNS ecosystem. However, simply getting the addresses of the authoritative servers, does not supply enough information to infer some kind of centralization. Further information is needed for this such as ASes and the AS number. Thus, we use a publicly available prefix-to-AS Table (FRANK, 2018) for each IPv4 address to obtain the corresponding AS of the hop, as well as the owner of the hop (*i.e.,* to determine that is company is responsible for running the infrastructure). This stage is repeated for both the last hop and the HBTL.

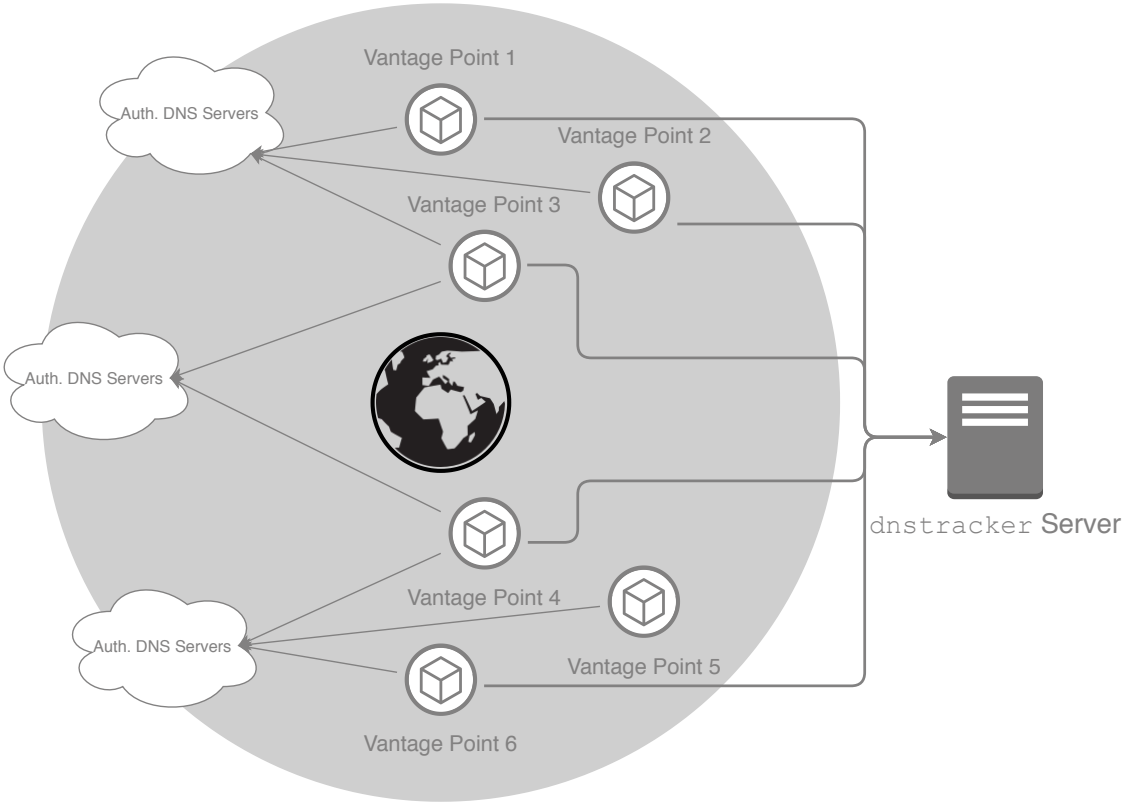
Finally, after the responses of all the hops as far as the targeted authoritative DNS servers, have been received and the corresponding ASes of each hop properly mapped, this information is stored in our database for further analysis. When this process is executed repeatedly, we are able to consolidate the millions of entries in the database and seek to

find out if there is a possible infrastructure aggregation at many different levels, as well as to analyze any changes in the DNS ecosystem over a period of time.

3.2 Vantage Points

Our solution that was set out in the 3.1 Section was designed to operate in a distributed way through collection agents (*i.e.,* to carry out a DNS collection of information from different vantage points). This is because more vantage points in different parts of the world can ensure a greater coverage of routes, and even different servers included in our solution.

Figure 3.3: Vantage points of our solution



Source: The Author

Figure 3.3 exemplifies how our solution was designed to handle with multiple vantage points. As can be seen, the different vantage points located around the world, can be used to collect routes for authoritative DNS servers from their location to the server. As a result of this distributed arrangement, we can obtain different routes from the vantage point to the server. Thus, we are able to obtain greater coverage of the possible query routes to the DNS servers.

3.3 How can the collected data be used?

Collecting information from the DNS infrastructure is only one part of our solution. On the basis of the data collection, an in-depth analysis can be carried out of several factors that may or may not help us to infer a possible centralization of the DNS infrastructure. In addition, the datasets generated from our solution can enable the DNS infrastructure to be characterized and show how it is displayed. With regard to the analysis of the centralization of the DNS infrastructure, we intend to analyze three facets which will be described below.

1. First, there is a need to evaluate the concentration of authoritative DNS servers per last hop AS. When conducting this analysis, we will be able to observe all the servers that share the infrastructure of a given provider so that they can host its authoritative DNS server. In this way it is possible to quantify the sites and services that rely on a single DNS service provider.
2. Second, we determine the total number of authoritative DNS servers that shared the same HBTL. Unlike the previous analysis and the analyses carried out in related works, when we analyse the concentration of authoritative DNS servers by HBTL we are increasing the granularity of our analysis. However it should be noted that there may be unique failures not only in the last hop (*i.e.,* the server itself) but also in a previous hop. The purpose of this analysis is to show providers that might be the cause of points of failure even before the last hop.
3. Third, we estimate the concentration of authoritative DNS server ASes per HBTL AS. So far, we have focused on analyzing the concentration of authorized servers in each hop. However, when looking for third-party ASes, other services may be affected, in addition to the hosted authoritative DNS servers. For this reason, we will also study the number of different ASes that share the same HBTL AS, in an attempt to detect shared points in the network infrastructure that may risk collateral damage for the authoritative DNS servers. In other words, a completely independent service might be the object of an attack and still affect the DNS ecosystem because of the shared infrastructure.

4 IMPLEMENTATION

This chapter, sets out the implementation of our solution. The `dnstracker` tool allows a case study to be undertaken and thus enables us to collect information related to DNS services and reveal the extent to which the DNS infrastructure has been centralized. The network operators and general users can view the concentration in a user-friendly web interface. In Section 4.1 below, we define our system architecture. In Section 4.2 and 4.3, we define some concepts about the tools `dig` and `traceroute`. In Section 4.4 we set out our server implementation. In Section 4.5 we describe the process of implementation of our agent. The source code for `dnstracker` is publicly available at GitHub[1].

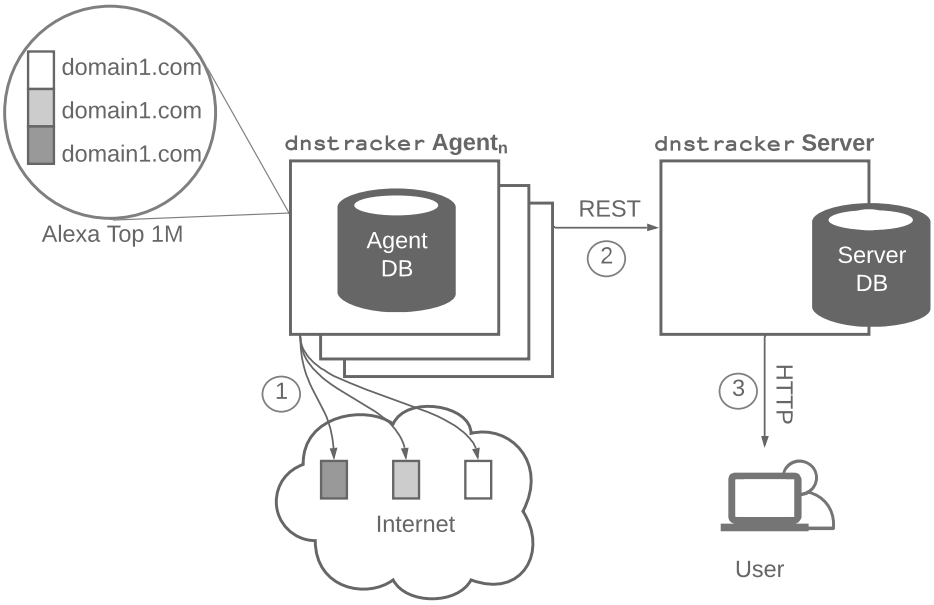
4.1 System Architecture

In this section, we describe the architecture of `dnstracker` on Figure 4.1:

1. **Agents** - On the left-hand side (1), a collection of `dnstracker` agents retrieve information, from targeted DNS authoritative servers by means of `traceroute`.
2. **Domain List** - The authoritative target servers are obtained from the list of the world's most popular websites, provided by Alexa Top 1 Million domain (list of open repositories) (ALEXA, 2018c), accessed in January 2018 (hosted as a local conceptual database inside each agent). The agent applies our solution for each domain in the list.
3. **Server** - After obtaining information from all the authoritative DNS servers, the `dnstracker` Agent exports the created datasets to the `dnstracker` Server (2) using a REST API (FIELDING, 2000).
4. **Web Interface** - After an export of the collected data from the `dnstracker` agent, the `dnstracker` server employs process tracing to create appropriate visualization and displays them for the users of the system via HTTP (3). We used the Spring Boot v2.0.4 framework to prototype this user interface[2].

[1] <https://github.com/ComputerNetworks-UFRGS/dnstracker>
[2] <http://dnstracker.inf.ufrgs.br>

Figure 4.1: `dnstracker` architecture



Source: The Author

4.2 Domain Information Groper - Dig

The *Domain Information Groper* - Dig is a flexible tool for querying DNS servers (CONSORTIUM, 2019a). Network administrators widely use it because of its simplicity. For example, it enables a list of authoritative DNS servers to be obtained for a DNS zone. It also creates queries for Type A records - which map a domain name to an IP address.

Figure 4.2: Disclosing authoritative DNS servers for the domain `ufrgs.br`

```
user@host:~$ dig ufrgs.br
; <<>> DiG 9.10.6 <<>> dig ufrgs.br
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 44828
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 3

;; AUTHORITY SECTION:
ufrgs.br.          1541    IN      NS      ns1.ufrgs.br.
ufrgs.br.          1541    IN      NS      ns2.ufrgs.br.
ufrgs.br.          1541    IN      NS      pampa.tche.br.

;; Query time: 8 msec
;; SERVER: 192.168.2.1
;; MSG SIZE rcvd: 98
```

Source: The Author

In our approach, we first run the *dig* command to each domain name in the list we run the dig command *e.g.* "*dig* `ufrgs.br`" to get the list of authoritative DNS servers to the `ufrgs.br` domain.

Figure 4.2 shows the return obtained when executing the command to obtain the authoritative servers for the given domain *e.g.* `ufrgs.br`. In the response, there is a header that indicates the type of server and its list. The authoritative DNS servers for the `ufrgs.br` domain are listed in the AUTHORITY SECTION (Figure 4.2).

After obtaining the list of authoritative DNS servers, it is possible to disclose Type A record of each of these servers using the same *dig* command. As can be seen in Figure 4.3, running the command *e.g.* "dig `ns1.ufrgs.br`" to one of the listed authoritative DNS servers in the **ANSWER SECTION** is a means of showing the Type A record of this server.

Figure 4.3: Disclosing Type A Record of the authoritative DNS server `ns1.ufrgs.br`

```
user@host:~$ dig ns1.ufrgs.br
; <<>> DiG 9.10.6 <<>> dig ns1.ufrgs.br
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 48185

;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0

;; ANSWER SECTION:
ns1.ufrgs.br. 600 IN A 143.54.1.58

;; Query time: 7 msec
;; SERVER: 192.168.2.1
;; MSG SIZE rcvd: 57
```

Source: The Author

In addition to *dig* there are other tools that can carry out the same discovery task as the domain servers, such as *NSLookup*. However, *NSLookup* is deprecated - as posted in a note from its development team (BERNSTEIN, 2019).

4.3 `Traceroute`

The `traceroute` tool is used by network administrators to perform the IP packet route tracking from the local server to a remote target server(CONSORTIUM, 2019b). To understand how the tool works, it is necessary to understand the operational techniques

of the TTL field, which can be found in the header of the IP datagrams.

Figure 4.4: `traceroute` to the `ns1.ufrgs.br` address

```
user@host:~$ traceroute -I ns1.ufrgs.br
traceroute to ns1.ufrgs.br (143.54.1.58), 64 hops max, 72 byte
    packets
 1  broadcom.home (192.168.25.1)  5.170 ms  3.518 ms  3.154 ms
 2  gvt-b-sr02.pae.gvt.net.br (179.184.126.60)  5.434 ms  5.144
    ms  5.010 ms
 3  191.30.9.225.dynamic.adsl.gvt.net.br (191.30.9.225)  5.393
   ms  5.085 ms  5.380 ms
 4  152-255-140-69.user.vivozap.com.br (152.255.140.69)  26.477
    ms  23.237 ms  24.027 ms
 5  as1916.saopaulo.sp.ix.br (187.16.216.4)  29.124 ms  21.801
   ms  22.089 ms
 6  200.143.255.141 (200.143.255.141)  34.244 ms  32.760 ms
   33.568 ms
 7  rs-sc-oi.bkb.rnp.br (200.143.252.57)  41.042 ms  41.551 ms
    43.860 ms
 8  * * *
 9  ufrgs-ve-40-mlxe8.tche.br (200.19.240.14)  57.957 ms
   55.861 ms  55.260 ms
10  143.54.0.249 (143.54.0.249)  51.498 ms  49.413 ms  50.506
   ms
11  lfs-in.ufrgs.br (143.54.0.241)  51.164 ms  50.613 ms
   52.429 ms
12  143.54.0.193 (143.54.0.193)  58.332 ms  56.246 ms  59.291
   ms
13  ns1.ufrgs.br (143.54.1.58)  54.862 ms  62.446 ms  63.371 ms
```

Source: The Author

The TTL (Time to Live) field, represents the maximum number of hops an IP packet can travel (*i.e.,* the maximum number of routers that will redirect a packet until it is dropped). The routers that implement the IP protocol subtract one unit from the value of the TTL field by one unit before routing each packet to its destination. If the TTL value reaches zero, the packet is discarded. The traceroute tool intelligently uses this parameter to trace all the routers involved in a packet's route to its destination server. The procedure adopted by the tool is as follows: the first packet sent by the tool to the destination server has the TTL field with a value of 1 in its header. Thus, the first router receiving this packet will decrement this value by 1 unit and accordingly discard the packet and respond to the source IP address. The second packet will be sent by the tool with the TTL field value 2, and so on consecutively, until it receives a response from the remote target server. By adopting this strategy, the traceroute tool can draw the route a packet travels to its

destination address. Figure 4.4 demonstrates the result of running the `traceroute` from a vantage point for one of the UFRGS authoritative DNS servers.

In our solution, we carry out a `traceroute` from a given vantage point for each authoritative DNS server obtained in the previous stage described in section 4.2. The `traceroute` provides information about the addresses of each hop in the route from the client to the domain's authoritative DNS server. Whenever a set of separate servers, owned by different websites, are hosted within the same provider's infrastructure, requests to these servers will share a common point - the network hop just before its final destination is reached, referred to as the *Hop-Before-The-Last* (HBTL). If two different requests are served by a path where the HBTL is in the same AS, they are likely to be hosted by the same DNS provider, and thus to some extent share the same infrastructure.

The `traceroute` tool is widely used for diagnosing network infrastructures. However, it does not ensure the parallelism necessary to provide reliable measurements across a large group of domains. The details of this problem and the consequent need to provide customization for the first tool are described in section 4.6.

4.4 `dnstracker`: Server

The server deployment was fully implemented in the Java programming language (JDK 8). The project manager Maven version 3.5.2 was also used for project planning and project management dependencies. When the project was first planned, it took into account the MVC (Model View Controller). This organization was abstracted by the Spring Boot framework, version 2.1.0. The MySQL database (version 5.7.25) was used to store the data generated by the collectors and Hibernate tool, version 5.3.1 for carrying out the object-relational mapping.

The `dnstracker` tool server exposes different REST routes which are used by a collection of `dnstrcker` agents to obtain and submit information related to their execution. This set of routes is also used by the WEB interface to obtain the data requested by the user for viewing purposes. The set of routes exposed by the server is examined in detail below:

- (GET - ***/api/versionInfo***) - Returns data from a pending collection run on a given agent. If no pending execution is found, a new execution identifier is generated. The data sent during this call are itemized in table 4.1.

- (GET - ***/api/resultData***) - Returns the resulting data for a given collection, following a given aggregation.
- (GET - ***/api/allAvailableRuns***) - Lists the details of all the pending runs.
- (GET - ***/api/dnsTrackerEntry***) - Gets an entry for collecting information about a DNS server based on its identifier.
- (GET - ***/api/processedDomains***) - Returns the list of domains already processed for a given collection run.
- (POST - ***/api/dnsTrackerEntries***) - Receives a collection of entries and saves them at the database, while updating the execution status. This is the route that the tool agents call on to send a batch of collection results. The data sent during this call are listed in Table 4.2.
- (POST - ***/api/versionInfo/:id/finish***) - Ends a particular collection run.

A record for each collection run, a record is stored in the database in a table `version_info` whose object-relational mapping is given in Table 4.1. This record is used to ensure and certify that all the collections have been completed successfully.

Table 4.1: Data Model - Execution of the Collection Instance

Field	Type	Description
id	Integer	Execution identifier
startDate	DateTime	Date and time of the start of the execution
endDate	DateTime	Date and time when the execution ended. This field is *null* if the collection has not yet been finalized.
region	Text	Vantage point identifier (used as the agent identifier).
workerIp	Text	IP address of the collection agent.

Source: The Author

During the collection processing for a given domain, a new database record is compiled for each authoritative DNS server. This record is stored in the table `domain_dnss`, and its object-relational mapping is given in table 4.2.

4.5 `dnstracker`: Agent

The `dnstracker` agent contains the core of the operational logic within the tool. It is responsible for obtaining data from authoritative DNS servers for a domain by tracking the route to these servers as well as by mapping the IP addresses obtained

Table 4.2: Data Model - Data Collected by DNS Server

Field	Type	Description
traceVersion	Integer	Identifier of the execution to which the record is bound.
domain	Text	Domain Name to which the DNS Server is attached.
position	Integer	Domain position in the Alexa Top 1 Million list.
nsName	Text	Authoritative DNS Server name.
nsIp	Text	Authoritative DNS Server IPv4 Address.
nsIpv6	Text	Authoritative DNS Server IPv6 Address.
nsAsn	Integer	Authoritative DNS Server Autonomous System Number.
nsSubnet	Text	Authoritative DNS Server Subnet Mask.
hbtlName	Text	Hop-Before-The-Last Name.
hbtlIp	Text	Hop-Before-The-Last IP Address.
hbtAsn	Integer	Hop-Before-The-Last Autonomous System Number.
hbtlSubnet	Text	Hop-Before-The-Last Subnet Mask.

Source: The Author

from Autonomous Systems. All these tasks are regarded as I/O-intensive. In view of this feature, the agent implementation must effectively exploit the parallelism techniques to reduce the total amount of time required for the collection.

The agent implementation was done entirely in the Java programming language using the Java Development Kit, version 8 (JDK 8). The project manager with Maven version 3.5.2 was used for project planning and dependency management. When the project was planned, it took into account the Model View Controller (MVC) standart for software architecture. This organization was abstracted by the Spring Boot framework, version 2.1.0. All communication between the agent and server is conducted through HTTP REST calls with data in the JSON format.

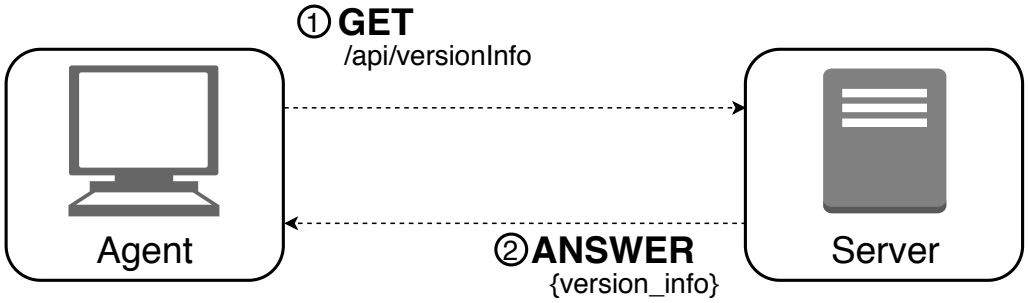
Each agent has a configuration file where it is possible to parameterize the agent identifier among other factors. This parameter is important since it allows the agent to be identified by the server, as well as the agents to be combined with different vantage points.

When the process is started, the agent is loaded into memory and initializes both databases that are required during its execution:

- *Alexa's List Top 1 Million*: The list is loaded through CSV file that is publicly available (ALEXA, 2018c).
- *prefix-to-AS Table*: On the basis of the announced prefixes, it is possible to determine which Autonomous System has an IP block that is combined with a given IP address (FRANK, 2018)

After this initialization, the agent sends a request to the server giving information to its identifier. On the basis of this, the server queries its database to determine if any collection is already running for this agent. If so, the server returns the agent to the list of domains that were already being processed during this execution. This kind of verification allows the resumption of a run that may have been interrupted by problems such as a server crash, agent crash and communication issues. If no pending execution is found, the server records the start of new execution, and returns the command to the agent to begin a new collection. This communication is illustrated in Figure 4.5.

Figure 4.5: API call to the route `/api/versionInfo`



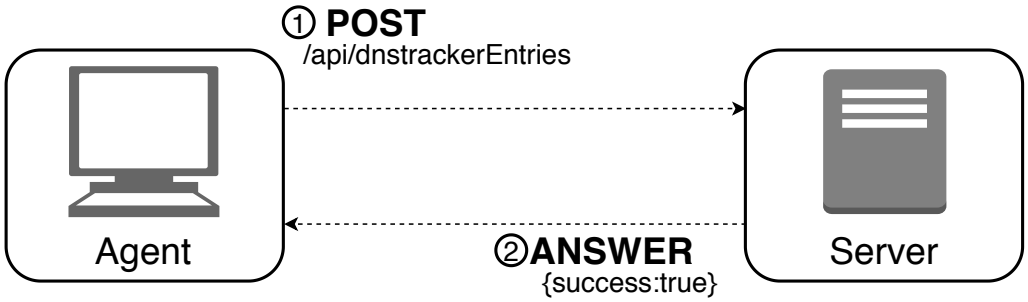
Source: The Author

When a new collection starts, a new thread pool of performers instances is created. These performers are given the task of collecting information for a given Fully Qualified Domain - FQDN. The size of this thread pool is parameterizable, and allows the collector agent agent to be scaled in line with the capacity of the server where it is allocated. If the parameter is not entered, it is initialized to the default value of 150 - that is, 150 separate domains will be processed simultaneously by default. Each of these performers executes the collection algorithm for a domain independently.

Initially, the collector obtains a domain name that must be processed on the basis of Alexa Top 1 Million list, which is loaded during startup. When obtaining the domain that must be processed, it is removed from the list to ensure that it is not processed in duplicate by the other collectors. This is done in an atomic transaction. After obtaining a domain name, the `dnstracker` agent executes a `dig command` to obtain the list of Authoritative DNS servers of this domain. Following this, the network route is traced, for each of the DNS servers found, by means of a custom version of the `traceroute` command. The IP addresses of two points of interest are extracted from this trace. The target DNS server and the network hop immediately precede the target DNS server (Hop-Before-The-Last). For both addresses, the prefix-to-AS table (FRANK, 2018) is used to

map the Autonomous Systems to which each one belongs. All this information is stored in a buffer, which will be sent to the `dnstracker` server after it reaches a predetermined size. This buffer is used to reduce the volume between the client and the server, and gathers together a certain number of records so that they can all be transmitted at once.

Figure 4.6: API call to the route `/api/dnsTrackerEntries`



Source: The Author

The collected data is then sent to the server through a call REST, illustrated in Figure 4.6. This process is repeated by the executor until all the domains have been processed. After the processing has been completed, the executors are terminated, and the agent sends a request to the server, and informs it of the successful completion of the collection operation.

4.6 Custom `traceroute`

The `traceroute` tool was initially designed to allow the network administrators to conduct a route analysis and assist in problem-solving. Its operating principle is based on sending ICMP Echo Requests.

However, the tool was not designed to be applied to problems that require simultaneous executions within the same process. In addition, owing to the way it was implemented, there was the risk of packet collision in the 64-bit operating systems, which could lead to a a mishandling of the responses received by the operating system, and hence errors in the interpretation of the data provided by the tool. The code snippet in Figure 4.7 is taken from the source code of the `traceroute` tool implemented in the FreeBSD operating system (MIT, 2018).

Line 4, shows the operation carried out by the tool to assign the identification field to the ICMP packets that must be sent. By only taking note of the least significant bits of the process identifier (through the 0xffff mask), the implementation opens up spaces

Figure 4.7: Traceroute tool code snippet

```
outip -> ip_dst = to -> sin_ addr;
outip -> ip_hl = (outp - (u_char*) outip) >> 2;

ident = (getpid() & 0xffff) | 0x8000 ;

if (pe == NULL) {
    Fprintf (stderr, "%s: unknown protocol%s \n ", prog, cp );
    exit(1) ;
}
```

Source: The Author

for the occurrence of identifier collisions in specific cases, as simulated by Code 4.8. In addition, the occurrence of collisions is evident in cases where multiple instances of the tool are executed in parallel by the same process.

Figure 4.8: ID collision simulation

```
#include <sys/types.h>
#include <unistd.h>

int main ( )
{
    pid_t pid1 = 5;
    pid_t pid2 = 65541;

    pid_tident1 = (pid1 & 0xffff) | 0x8000 ;
    pid_tident2 = (pid2 & 0xffff) | 0x8000 ;
    printf ("Packet ID1: %d\nPacket ID2: %d", ident1, ident2);
    return 0;
}
```

Source: The Author

Figure 4.8 shows the code written in C language that simulates the calculation of the identifier that is carried out by the `traceroute` tool in a 64-bit operating system, in a hypothetical scenario where the tool is being executed by two processes simultaneously. In this scenario, these types of processes have PIDs 5 and 65541. In the simulation, the identifiers generated by the `traceroute` tool are the same, which shows the occurrence of a collision.

In tackling this issue, it was necessary to implement a custom version of the `traceroute` tool that does not have the same limitations caused by concurrent executions. Since the Java programming language does not have native support for low-level

Figure 4.9: Result of executing the code shown in Figure 4.8

```
user@localhost - ~$ ./colision_simulator

Packet ID1: 32773
Packet ID2: 32773

...Program finished with exit code 0
Press ENTER to exit console.
```

Source: The Author

socket handling (owing to the particular features of each operating system) it was decided to use the RockSaw open-source JNI library (SAVARESE, 2018). This library provides a low-level interface for the native operating system sockets that are required to send out the ICMP packets. With the aid of the library, it was possible to reproduce the same functionality of the `traceroute` tool. This involved implementing the RawTraceRoute class, which enabled, the original `traceroute` to be reproduced by using the native interface of the library for the sockets and providing additional support for parallel executions. By instantiating the class, it is possible to enter the identifier that will be used by the package, as shown in Figure 4.10. In the `dnstracker` agent implementation, each thread is created by means of on a single incremental identifier, which is passed on as the identifier to the generated ICMP packet, and thus overcomes the problem of tag collision.

Figure 4.10: Code snippet of a *RawTraceRoute* class initialization

```
public RawTraceRoute(int id, int protocolFamily, int protocol)
   {
    this.sequence = 0;
    this.identifier = id;
    this.protocolFamily = protocolFamily;
    this.protocol = protocol;
}
```

Source: The Author

4.7 `dnstracker`: Web Interface

A web interface was designed with the aim of allowing easy access to the collection results obtained by the `dnstracker`, and this communicates with the `dnstracker` Server so as to be able to view the data. The interface was implemented in JavaScript,

by establishing the React framework, version 16.6. The Bootstrap framework, version 4 was used for creating the Windows and Maven Project manager version 3.5.2 for project organization and dependency management, Maven project manager version 3.5.2 was used. When the project was planned, it took into account the MVC (Model View Controller) software architecture standard. This abstraction was performed by the Spring Boot framework, version 2.1.0. In addition, the Tomcat WEB server, version 8 was used, which included Spring Boot.

Figure 4.11: `dnstracker` home page

Source: The Author

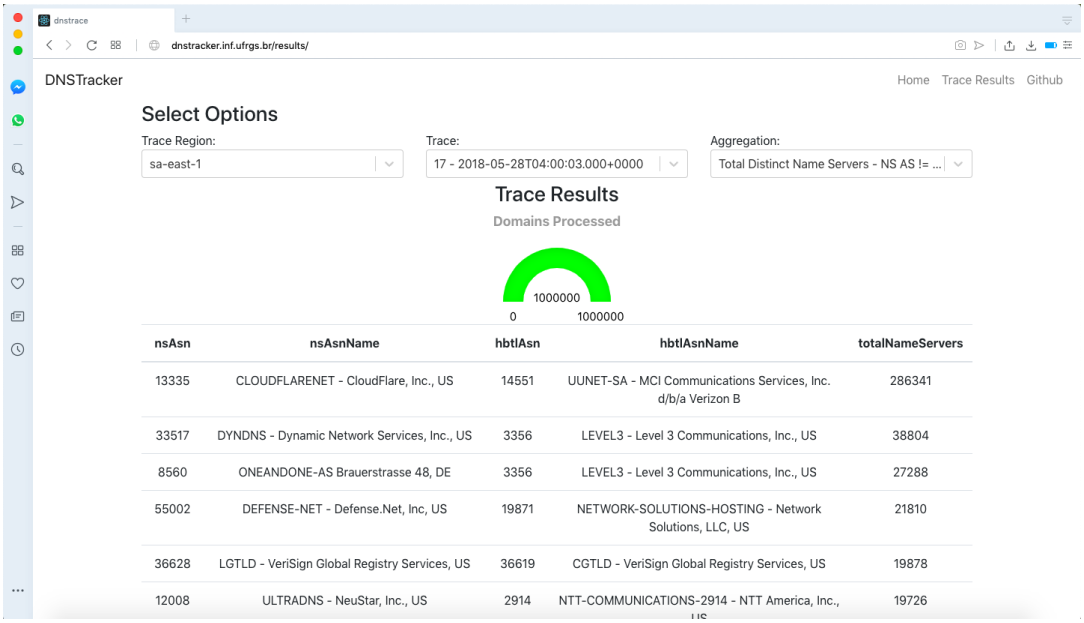
The `dnstracker` tool home page is displayed in Figure 4.11. It contains an introduction to the operating and developed tools and provides a means of accessing the results as well as using the source code of the tool at GitHub[3].

The user must click the *View Trace Results* button to access the results page. Then a screen will appear where the user has to choose one of the following options:

- **Trace Region**: The region from which the collections were made. This is the identifier of the agent responsible for the collection.
- **Trace**: The collection itself. The collections are sorted at the start time of the execution.
- **Aggregation**: The aggregation for which the data must to be processed and displayed.

[3]<https://github.com/ComputerNetworks-UFRGS/dnstracker>

Figure 4.12: Results page of `dnstracker`



Source: The Author

On the basis of these choices, a request is sent to the server a) for the search of the database, b) to carry out the aggregate processing, and c) for the return data to the interface. A Table is displayed with the results as well as an indicator of how many domains have already been processed during the chosen collection run.

5 RESULTS

This chapter examines the results obtained from our solution. Our research drew on the data collection generated by the `dnstracker` tool. Three factors are investigated to determine if there is a possible infrastructure centralization or possible collateral damage caused by infrastructure sharing. Below, we present the datasets generated by our solution in Section 5.1. In the following sections we present our results.

5.1 Datasets

Our solution was employed several times a month, from January 2018 to May 2018, and resulted in a dataset of millions of *traceroute* entries. Table 5.1 provides a summary of the data collected throughout the five months of observations. During the measurement period, the number of separate authoritative DNS servers, ASes, and HBTL ASes remained stable. In our samples, 136,421 out of the traced authoritative servers had ASes in their routes that openly rejected ICMP echo requests, which prevented information from being obtained about the HBTL of these servers. In light of this, these authoritative DNS servers were disregarded during the analysis of HBTL aggregation; this area was circumvented since it will be a subject of future work in our research.

Table 5.1: Datasets generated by `dnstracker` for monthly measurements of Alexa 1 million domain names.

	DNS data			`traceroute` data	
Month	NS rec.	IPv4 (NS)	Last Hop ASes	HBTL IPv4	HBTL ASes
Jan	283,983	208,543	18,400	40,157	7,742
Feb	283,983	208,543	18,400	40,157	7,742
Mar	283,983	208,543	18,400	40,157	7,742
Apr	283,983	208,543	18,400	40,157	7,742
May	283,983	208,543	18,400	40,157	7,742

Source: The Author

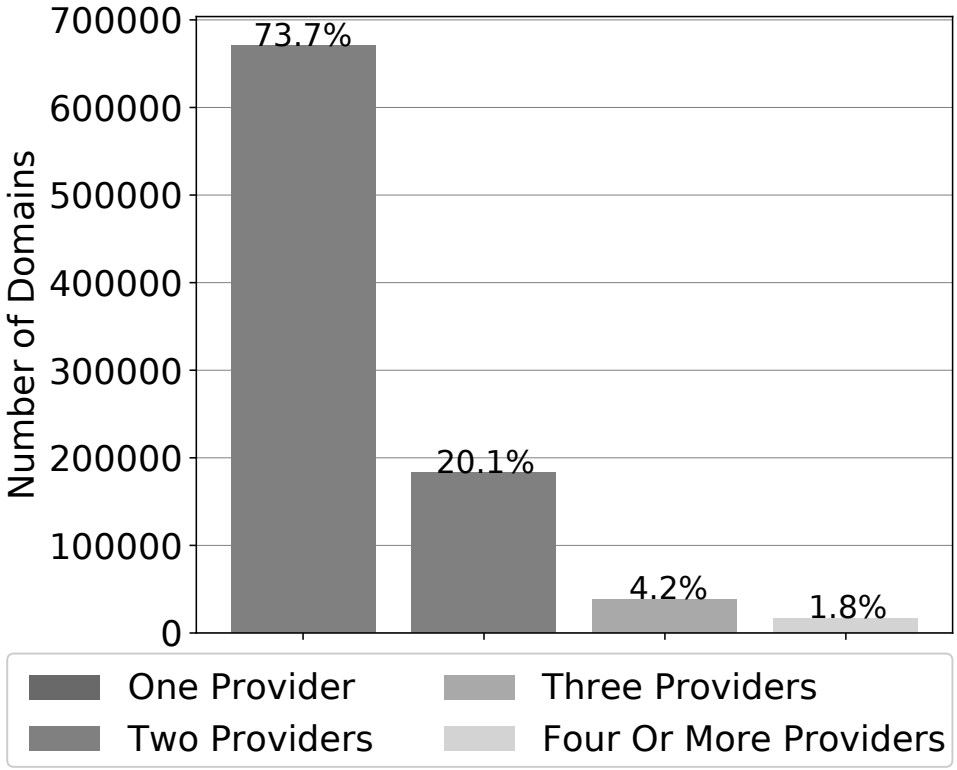
Having obtained the dataset through `dnstracker` , we concentrated on three factors to determine the degree of infrastructure centralization and possibility of collateral damage. First, we estimated the number of authoritative DNS servers per last hop AS. Second, we measured the concentration of authoritative server ASes per HBTL AS. Third, we calculate the total number of authoritative DNS servers that shared the same HBTL.

These three calculations enabled us to measure the number of authoritative DNS servers that share their AS infrastructure with other authoritative DNS servers, at both last hop and HBTL level. Finally, we analyzed whether there is any growth trend in these aggregations among the top DNS providers during the measurement period. Each of these factors is explored in the following sections.

5.2 DNS Space Diversification

The concentration of the DNS space can have the potential to expose dangerous single points of failure in the DNS ecosystem. Despite this, a valuable problem prevention practice is to register multiple DNS servers, including the use of multiple DNS providers. For example, to a domain, *i.e.,* `www.example.com`, you can register more than one DNS server with authority for that domain and each name server can be managed by a different DNS provider.

Figure 5.1: Domains that registered one, two, three or more DNS providers



Source: The Author

RFC 2182 describes the recommended practices for selecting secondary DNS

servers. This RFC explains that one of the main reasons for having multiple servers for each zone is to allow the zone's information to be available in a wide and reliable manner (ELZ et al., 1997). Likewise, when choosing to use multiple providers, a domain can ensure that it will have redundancy and robustness, even in an increasingly concentrated DNS space.

Our analysis showed that most domains are not taking advantage of the possibility of diversifying DNS providers. Figure 5.1 shows the percentage of domains in our sample using name servers with one provider was 73.7%. This means that most of the domains analyzed do not make use of DNS server redundancy. Use two DNS providers prevents a domain from being inaccessible because it is very difficult for two providers to be offline at the same time. The number of domains using two DNS providers drops dramatically to 20.1%.

5.3 Aggregation of Authoritative DNS Servers by Last Hop ASes

We analyzed the concentration of different authoritative DNS servers by means of the last hop ASes. Table 5.2 and Figure 5.2 show the top 10 ASes that aggregated the most authoritative DNS servers. For display purposes, each AS set out in Table 5.2 is represented in Figure 5.2 identified by its name. In addition, in Table 5.2 each line shows the name and number of each AS. As can be seen, most of the ASes that were found belong to big infrastructure providers.

Table 5.2: Authoritative DNS Server Aggregation by Last Hop ASes

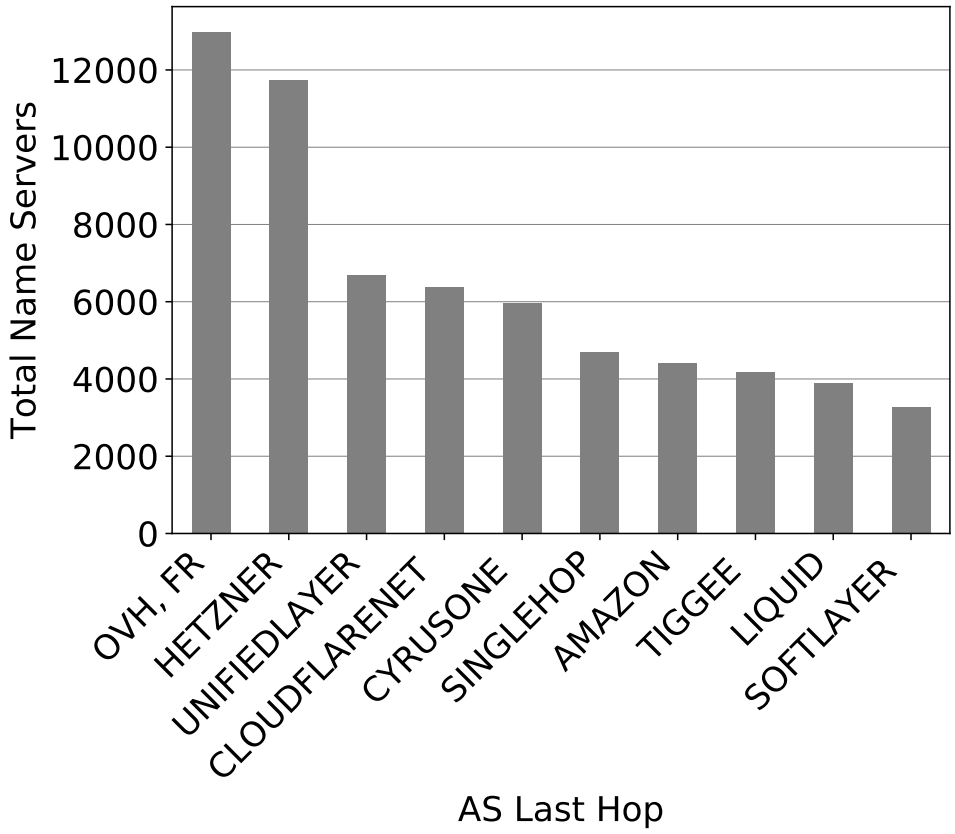
ID	AS Name	Auth. NS	AS Number
AS1	OVH, FR	12.990	16.276
AS2	HETZNER-AS, DE	11.730	24.940
AS3	UNIFIEDLAYER-AS-1 - Unified Layer, US	6.698	46.606
AS4	CLOUDFLARENET - CloudFlare, Inc., US	6.384	13.335
AS5	CYRUSONE - CyrusOne LLC, US	5.955	20.013
AS6	SINGLEHOP-LLC - SingleHop, Inc., US	4.710	32.475
AS7	AMAZON-02 - Amazon.com, Inc., US	4.421	16.509
AS8	TIGGEE - Tiggee LLC, US	4.182	16.552
AS9	LIQUID-WEB-INC - Liquid Web, L.L.C, US	3.890	32.244
AS10	SOFTLAYER - SoftLayer Technologies, US	3.265	36.351

Source: The Author

The figure shows that the "OVH, FR" provider, indicated by the `AS1` identifier

in Table 5.2, is the DNS provider with the largest number of authoritative servers in its infrastructure, and aggregates a total of more than 12,000 different authoritative servers, each with multiple hosted domains. This means that if there was a successful attack on this AS, over 77,000 websites would be unreachable, since the clients would not be able to resolve the FQDNs.

Figure 5.2: Authoritative DNS Server Aggregation by Last Hop AS



Source: The Author

Thus, it can be seen that the "HETZNER-AS, DE" provider, designated as `AS2`, holds 11,000 of the total authoriative servers, which represents 30,000 distinct websites hosted, followed by the `AS3` "UNIFIEDLAYER-AS-1 - Unified Layer, US", which hosts 6,000 authoritative servers within its infrastructure representing 6,000 distinct websites. However these top 3 ASes incur a high risk of collateral damage, as they concentrate a large number of authoritative DNS servers. `AS4` to `AS10`, there is a margin ranging from 6,000 to 3,000 of the total number of DNS servers in each of the providers. This level of centralization has been pointed out in previous studies (ALLMAN, 2018) through an examination of the IP blocks of the authoritative servers, wich show how many separate authoritative DNS servers belonged to the same IP block. Our study follows the pattern

of previous studies (BATES et al., 2018; ALLMAN, 2018), by underlining the degree of centralization in DNS services, which can lead to collateral damage, as already mentioned (VIXIE; SNEERINGER; SCHLEIFER, 2002; OPERATORS, 2015; OPERATORS, 2016; WEINBERG M., 2016; MOURA et al., 2016; SENGUPTA, 2012). It requires a good deal of technical ingenuity to make an infrastructure of this size fail. However, it should be noted that DDoS attacks are becoming increasingly sophisticated, including the capacity to seize control of the infrastructure of DynDNS (HILTON, 2016), so this concern deserves serious attention.

5.4 Aggregation of Authoritative DNS Servers by HBTL ASes

In addition to analyzing the amount of shared infrastructure in last hop ASes by authoritative server, we inspected the number of authoritative DNS servers that share the same HBTL, since there might only be a single point of failure. Table 5.3 and Figure 5.3 display the top 10 HBTL ASes that aggregated the most authoritative DNS servers. For display purposes, each AS shown in Table 5.3 is represented in Figure 5.3 identified by its name, just as in the last analysis. In addition, each line shows the name and number of each ASes. As can be seen, most of the found ASes belong to big infrastructure providers as well.

Table 5.3: Authoritative Name Server Aggregation by HBTL

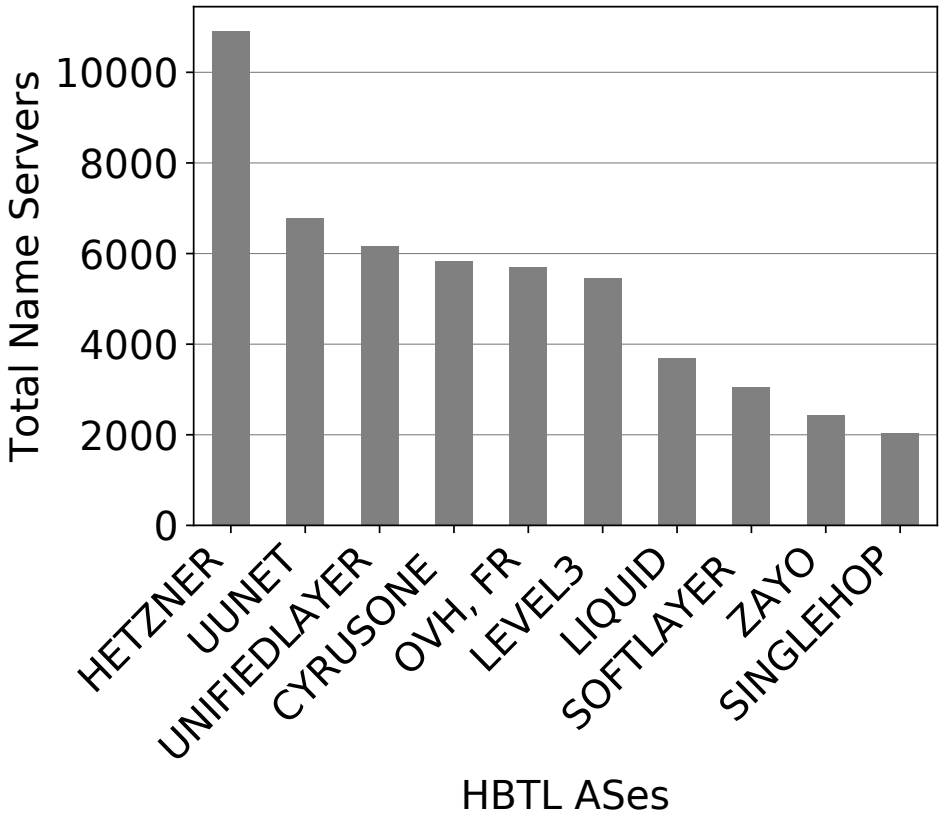
ID	AS Name	Auth. NS	AS Number
AS1	HETZNER-AS, DE	10.904	24.940
AS2	UUNET-SA - MCI Communications Services	6.789	14.551
AS3	UNIFIEDLAYER-AS-1 - Unified Layer, US	6.173	46.606
AS4	CYRUSONE - CyrusOne LLC, US	5.826	20.013
AS5	OVH, FR	5.708	16.276
AS6	LEVEL3 - Level 3 Communications, Inc., US	5.458	3.356
AS7	LIQUID-WEB-INC - Liquid Web, L.L.C, US	3.683	32.244
AS8	SOFTLAYER - SoftLayer Technologies Inc., US	3.043	36.351
AS9	ZAYO-6461 - Zayo Bandwidth Inc, US	2.442	6.461
AS10	SINGLEHOP-LLC - SingleHop, Inc., US	2.037	32.475

Source: The Author

`AS1`, which is designated as "HETZNER-AS, DE" in Table 5.3, shows that almost

11,000 of the total authoritative servers share the same hop as its HBTL. We mention that HBTL may change depending on the vantage point. Other vantage points will be analyzed in future work. The "UUNET-SA - MCI Communications Services, Inc.", represented by the `AS2` identifier, is shared by almost 7,000 authoritative servers as well. These numbers suggest the presence of centralization in the DNS infrastructure itself, not only at the last hop, as mentioned by previous studies (BATES et al., 2018)(ALLMAN, 2018), but also in the HBTL as well. In addition, it should be stressed once more that each of these authoritative servers resolve thousands of domains. Hence, if an HBTL was to be taken down, hundreds of thousands of domains would become unreachable as result of collateral damage.

Figure 5.3: Authoritative Name Server Aggregation by HBTL



Source: The Author

5.5 Aggregation of Last Hop ASes by HBTL ASes

So far, this study has focused on analyzing the concentration of authoritative servers in each hop. However, when looking for a third-party provider ASes, other services may be affected as well as the hosted authoritative DNS servers. Hence, we also examined the number of different ASes that share the same HBTL AS, with the aim of detecting points in the shared network infrastructure that might incur a risk of collateral damage to authoritative DNS servers, *i.e.,* a completelly unrelated service might be targeted by an attack and still affect the DNS ecosystem because of its shared infrastructure. Once again, Table 5.4 and Figure 5.4 show the top 10 HBTL ASes where the largest number of last hop ASes are concentrated. For display purposes, each AS shown in Table 5.3 is represented in Figure 5.3 identified by its name. In addition, each line includes the name and number of each AS. As can be seen, most of the found ASes belong to large infrastructure and network providers as well, such as Level 3 and Cogent.

Table 5.4: Last Hop AS Aggregation by HBTL AS

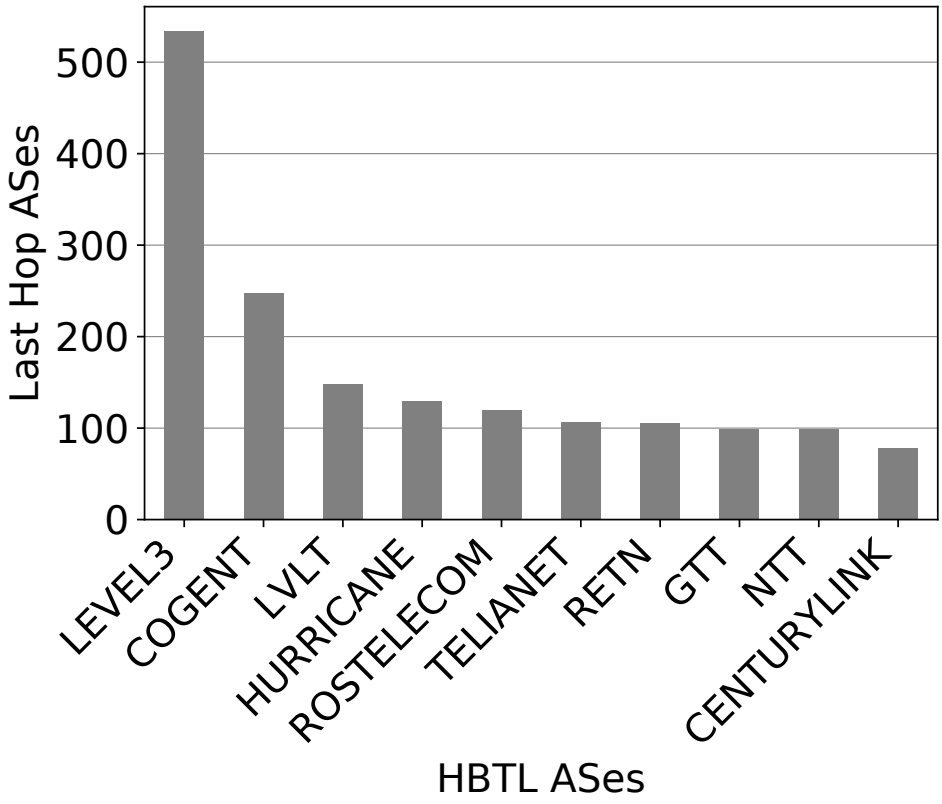
ID	AS Name	ASes	AS Number
AS1	LEVEL3 - Level 3 Communications, Inc., US	534	3.356
AS2	COGENT-174 - Cogent Communications, US	248	174
AS3	LVLT-3549 - Level 3 Communications, Inc., US	148	3.549
AS4	HURRICANE - Hurricane Electric, Inc., US	130	6.939
AS5	ROSTELECOM-AS, RU	120	12.389
AS6	TELIANET Telia Carrier, SE	107	1.299
AS7	RETN-AS, UA	106	9.002
AS8	GTT-BACKBONE GTT, DE	99	3.257
AS9	NTT-COMMUNICATIONS-2914	99	2.914
AS10	CENTURYLINK-US-LEGACY-QWEST	78	209

Source: The Author

In this assessment, the most noteworthy aggregation of last hop ASes occurs in "LEVEL3 - Level 3 Communications, Inc., US" HTBL AS, identified as `AS1`. Level 3 is one of the leading top infrastructure providers in the world, so this is a natural result. However, the number of last hop ASes that share its infrastructure is large, and amounts to over 500 different ASes. The second largest HBTL aggregation provider - "COGENT-174 - Cogent Communications, US", designated as `AS2`, has less than half of the amount of Level 3, with 200 AS close behind it. Although the concentration of ASes behind a

single hop has probably more to do with delivery structure than DNS services, this kind of concentration increases the chance of problems to a larger number of service if targeted by a large-scale attack.

Figure 5.4: AS Aggregation by HBTL

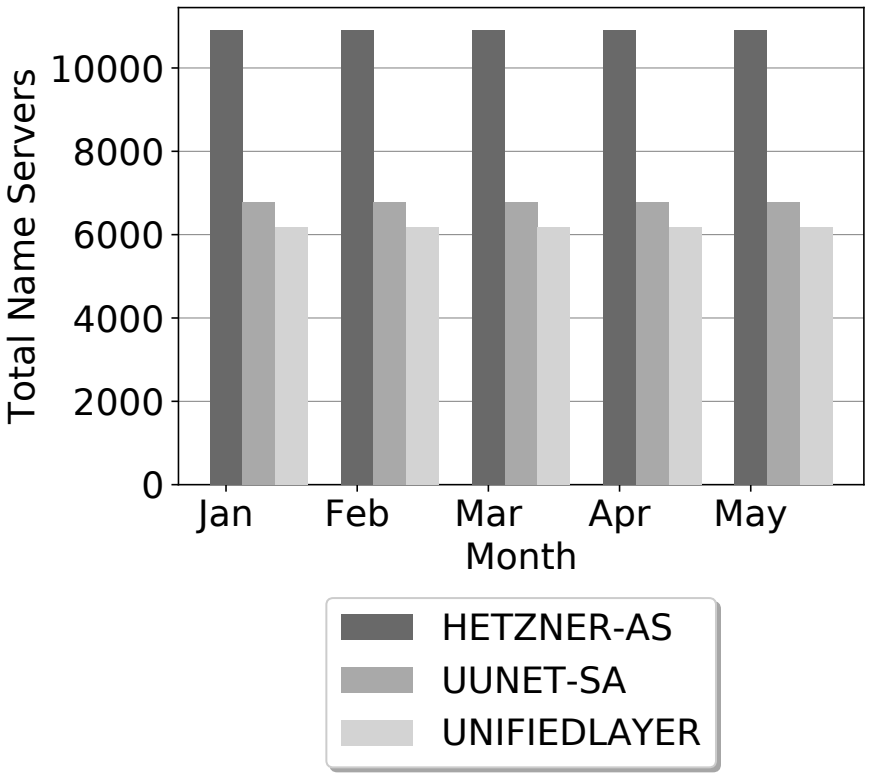


Source: The Author

5.6 Centralization Trend

Finally, as we measured DNS aggregation over a period of 5 months, there is a need to look at the difference in HBTL aggregation between the authoritative DNS servers. This can enable us to determine whether there is a centralizing trend in the DNS infrastructure, at least during this period. Figure 5.5 shows the aggregation level of the top 3 HBTL ASes, as found for each month when we traced the authoritative DNS servers.

Figure 5.5: Authoritative DNS Server aggregation by HTBL ASes over time



Source: The Author

The temporal graph shows that the centralization of authoritative DNS servers of the Alexa Top 1 Million websites remained stable in the period. This is consistent with the general assumption that the DNS infrastructure is stable and robust. In addition, it can be explained by the fact that the observed providers are reliable and there is no need for frequent changes when hosting them. However, this does not mean that there is no centralization trend when a larger time window is included.

6 CONCLUDING REMARKS

DNS plays a vital role in human interaction with the Internet. It translates human-friendly names into computer-readable information. As one of the essential services for the Internet to function, authoritative DNS servers have been the frequent victims of Distributed Denial-Service (DDoS) attacks. These DNS servers have been targeted several times in the last years. To counter these attacks, security measures were designed on the DNS infrastructure.

Although these measures are widely employed, when domain names share the same DNS provider, they may be sharing different levels of infrastructure. Because many companies do not run their DNS infrastructure instead of outsourcing to third-party DNS providers, third-party DNS providers may experience collateral damage in the event of an attack on the contracted provider. The Dyn (HILTON, 2016; NEWMAN, 2016) attack exemplifies collateral damage when authoritative DNS servers hosting multiple DNS zones are under attack.

In this context, in this dissertation, we present a flexible solution that makes it possible to evaluate the degree of centralization of authoritative DNS servers through active DNS measurements. Our study focused on analyzing a possible concentration of authoritative DNS servers in nature, for FQDNs. Also, we designed `dnstracker`, an open-source tool that implements our proposed solution and assists in consolidating our findings. As a case study, we use `dnstracker` to analyze all domains on the sites of the list of Alexa 1 Million.

6.1 Summary of Contributions

In this work, we present a solution that allows measuring, to various degrees, the level of centralization of authoritative DNS servers using active DNS measurements. In particular, we focus our work on analyzing a possible concentration on authoritative wild-type DNS servers for FQDNs. As a result of our approach, we identified possible "single point of failure" centralizations in some DNS providers. As a result of this shared infrastructure, there is a possibility of collateral damage in the event of a successful attack on one of these providers.

We present the implementation of a tool that performs the collection of information from routes to the authoritative servers for a set of domain names. Besides, it can

aggregate data in a web interface to facilitate the use of the information by network operators or people interested in verifying centralization issues of the DNS infrastructure. Our experiences have shown that the tool enabled the collection of information necessary to carry out an analysis of the centralization of the DNS infrastructure.

As a case study, we used `dnstracker` to analyze all domains of the Alexa Top 1 Million (ALEXA, 2018c) websites. The analysis that was conducted entailed a considerable amount of infrastructure sharing, at many different levels of the DNS ecosystem. We show that, in some cases, up to 12,000 authoritative name servers of the most visited websites share the same infrastructure of big DNS providers, and thus could suffer from collateral damage in the event of an attack. The level of concentration of authoritative DNS servers by HBTL also poses a risk of collateral damage, as most DNS operators are not aware of this kind of concentration when contracting hosting services. On the other hand, if one only looks at the last hop ASes for concentration, it may be misleading because many companies (*e.g.* `facebook.com`) may advertise their ASes for their authoritative servers, but still rely on the infrastructure of a third-party provider. In cases of this kind, the possibility of collateral damage remains, although it has so far been undetected.

We also analyzed the diversity of the DNS space. We show that in more than 70% of the domains evaluated, do not use techniques to increase the guarantee of the availability of their services, such as the use of more than one DNS provider. In this way, these domains depend only on a single DNS provider and may suffer damage in the event of a successful DDoS attack. Besides, we analyzed our measurements collected during five months to try to identify a centralization trend in the DNS global infrastructure. However, no trend was identified in the period we collected our data.

6.2 Final Remarks and Future Work

The future directions of our research include observing DNS centralization from different vantage points on the Internet; we want to understand how vital a vantage point is in our observation solution. The use of more vantage points should enable an enrichment in the routes and also the discovery of different paths for each server; that way, we can carry a more in-depth analysis. We also intend to improve our collection solution about some points of observation that can be developed. Among these is the collection of all hops to an authoritative DNS server instead of collecting only the last two. We

understand that we need to deepen this issue to conduct a more in-depth analysis of the global DNS infrastructure. Finally, at a more theoretical perspective, we are working on a centralization metric that will help network operators find the more appropriate hosts for their DNS needs.

REFERENCES

1.1.1.1. *The Internet's Fastest, Privacy-First DNS Resolver*. 2018. <https://1.1.1.1/>. Accessed 25 Aug. 2018.

ABHISHTA, A.; RIJSWIJK-DEIJ, R. van; NIEUWENHUIS, L. J. Measuring the impact of a successful DDoS attack on the customer behaviour of managed DNS service providers. **ACM SIGCOMM Computer Communication Review**, ACM, v. 48, n. 5, p. 70–76, 2019.

AGER, B. et al. Comparing DNS resolvers in the wild. In: ACM. **Proceedings of the 10th ACM SIGCOMM conference on Internet measurement**. [S.l.], 2010. p. 15–21.

ALEXA. **The Alexa Extension**. 2018. <https://web.archive.org/web/20160604100555/http://www.alexa.com/toolbar.>. Accessed 25 Aug. 2018.

ALEXA. **Alexa Increases its Global Traffic Panel**. 2018. <https://blog.alexa.com/alexa-panel-increase/.>. Accessed 25 Dec. 2019.

ALEXA. **Top 1M sites. https://www.alexa.com/topsites**. 2018. <http://s3.dualstack.us-east-1.amazonaws.com/alexa-static/top-1m.csv.zip.>. Accessed 1 Jan. 2018.

ALEXA. **Top 6 Myths about the Alexa Traffic Rank**. 2018. <https://blog.alexa.com/top-6-myths-about-the-alexa-traffic-rank/.>. Accessed 25 Dec. 2019.

ALLMAN, M. Comments on DNS Robustness. In: **ACM Internet Measurement Conference**. [S.l.: s.n.], 2018.

ARMIN, J. et al. 2020 cybercrime economic costs: No measure no solution. In: IEEE. **2015 10th International Conference on Availability, Reliability and Security**. [S.l.], 2015. p. 701–710.

ATTACK, D. A. S. O. F. O. . **https://dyn.com/blog/dyn-analysis-summary-of-friday-october-21-attack/**. 2018. <https://dyn.com/blog/dyn-analysis-summary-of-friday-october-21-attack.>. Accessed 1 Jan. 2018.

BATES, S. et al. **Evidence of Decreasing Internet Entropy: The Lack of Redundancy in DNS Resolution by Major Websites and Services**. [S.l.], 2018.

BERNSTEIN, D. J. **Notes on NSlookup**. 2019. <http://cr.yp.to/djbdns/nslookup.html.>. Accessed 31 Dez. 2019.

BILGE, L. et al. Exposure: A Passive DNS Analysis Service to Detect and Report Malicious Domains. **ACM Transactions on Information and System Security (TISSEC)**, ACM, v. 16, n. 4, p. 14, 2014.

CHANDRAMOULI, R.; ROSE, S. Challenges in securing the domain name system. **IEEE Security & Privacy**, IEEE, v. 4, n. 1, p. 84–87, 2006.

CISCO. 2017. <https://umbrella.cisco.com/blog/2014/07/16/difference-authoritative-recursive-dns-nameservers/.>. Accessed 1 Jan. 2018.

CONSORTIUM, I. **https://linux.die.net/man/1/dig**. 2019. <https://linux.die.net/man/1/dig.>. Accessed 05 Feb. 2020.

CONSORTIUM, I. **https://linux.die.net/man/8/traceroute**. 2019. <https://linux.die.net/man/1/traceroute.>. Accessed 05 Feb. 2020.

DEUTSCH, L. **Host names on-line**. [S.l.], 1973. <http://www.rfc-editor.org/rfc/rfc606.txt".>. Accessed 24 Dec. 2020.

ELZ, R. et al. **Selection and Operation of Secondary DNS Servers**. 1997. <http://tools.ietf.org/rfc/rfc2182.txt.>. Accessed 13 Apr. 2019.

FIELDING, R. T. **Architectural Styles and the Design of Network-based Software Architectures**. Thesis (PhD), 2000. University of California, Irvine.

FILIPPI, P. D.; MCCARTHY, S. Cloud computing: Centralization and data sovereignty. **European Journal of Law and Technology**, v. 3, n. 2, 2012.

FOMENKOV, M. et al. Macroscopic internet topology and performance measurements from the dns root name servers. In: **LISA**. [S.l.: s.n.], 2001. p. 231–240.

FRANK, D. **Free IP address to ASN database**. 2018. <https://iptoasn.com/.>. Accessed 17 Nov. 2019.

HILTON, S. **Dyn Analysis Summary Of Friday October 21 Attack**. 2016. Dyn blog <https://dyn.com/blog/dyn-analysis-summary-of-friday-october-21-attack/>. Accessed 25 Aug. 2018.

KIM, W. et al. The dark side of the internet: Attacks, costs and responses. **Information systems**, Elsevier, v. 36, n. 3, p. 675–705, 2011.

KUROSE, J. F. **Computer networking: A top-down approach featuring the internet, 3/E**. [S.l.]: Pearson Education India, 2005.

LIU, C.; ALBITZ, P. **DNS and Bind**. [S.l.]: " O'Reilly Media, Inc.", 2006.

LIU, Z. et al. Two days in the life of the dns anycast root servers. In: SPRINGER. **International Conference on Passive and Active Network Measurement**. [S.l.], 2007. p. 125–134.

MCPHERSON, D. et al. **Architectural Considerations of IP Anycast**. 2014. <http://tools.ietf.org/rfc/rfc7094.txt.>. Accessed 20 Dec. 2019.

MOCKAPETRIS, P. **Domain names: Concepts and facilities**. [S.l.], 1983. <http://www.rfc-editor.org/rfc/rfc882.txt.>. Accessed 1 Jan. 2020.

MOCKAPETRIS, P. **Domain names: Implementation specification**. [S.l.], 1983. <http://www.rfc-editor.org/rfc/rfc883.txt.>. Accessed 1 Jan. 2020.

MOCKAPETRIS, P. **Domain names - concepts and facilities**. [S.l.], 1987.

MOCKAPETRIS, P. **Domain names - implementation and specification**. [S.l.], 1987. <http://www.rfc-editor.org/rfc/rfc1035.txt.>. Accessed 1 Jan. 2020.

MOCKAPETRIS, P. V.; DUNLAP, K. J. Development of domain name system. 1988.

MOURA, G. C. M. et al. **Anycast vs. DDoS: Evaluating the November 2015 Root DNS Event**. 2016.

MUGALI, A. A. et al. **System and method for detecting DNS traffic anomalies**. [S.l.]: Google Patents, 2015. US Patent 9,172,716.

NEWMAN, L. H. **What We Know About Friday's Massive East Coast Internet Outage**. 2016. <https://www.wired.com/2016/10/internet-outage-ddos-dns-dyn/>. Accessed 1 Jan. 2020.

OPERATORS, R. S. **Events of 2015-11-30**. 2015. <http://root-servers.org/news/events-of-20151130.txt>. Accessed 25 Aug. 2018.

OPERATORS, R. S. **Events of 2016-06-25**. [S.l.], 2016. <http://www.root-servers.org/news/events-of-20160625.txt>. Accessed 25 Aug. 2018.

PERDISCI, R.; CORONA, I.; GIACINTO, G. Early detection of malicious flux networks via large-scale passive dns traffic analysis. **IEEE Transactions on Dependable and Secure Computing**, IEEE, v. 9, n. 5, p. 714–726, 2012.

PERLROTH, N. Hackers used new weapons to disrupt major websites across U.S. p. A1, Oct. 22 2016. <http://www.nytimes.com/2016/10/22/business/internet-problems-attack.html>. Accessed 25 Aug. 2018.

POSTEL, J. **Computer mail meeting notes**. [S.l.], 1982. <http://www.rfc-editor.org/rfc/rfc805.txt>. Accessed 25 May. 2018.

RIJSWIJK-DEIJ, R. M. van. **Improving DNS security: a measurement-based approach**. [S.l.]: University of Twente, 2017.

RIJSWIJK-DEIJ, R. van et al. A high-performance, scalable infrastructure for large-scale active dns measurements. **IEEE Journal on Selected Areas in Communications**, IEEE, v. 34, n. 6, p. 1877–1888, 2016.

SAHARAN, S.; GUPTA, V. Prevention and mitigation of dns based ddos attacks in sdn environment. In: IEEE. **2019 11th International Conference on Communication Systems & Networks (COMSNETS)**. [S.l.], 2019. p. 571–573.

SAVARESE, S. R. **RockSaw Raw Socket Library for Java**. 2018.

SCHEITLE, Q. et al. A long way to the top: significance, structure, and stability of internet top lists. In: ACM. **Proceedings of the Internet Measurement Conference 2018**. [S.l.], 2018. p. 478–493.

SENGUPTA, S. **After Threats, No Signs of Attack by Hackers**. 2012. <http://www.nytimes.com/2012/04/01/technology/no-signs-of-attack-on-internet.html>. Accessed 25 Dec. 2018.

SPEROTTO, A.; TOORN, O. van der; RIJSWIJK-DEIJ, R. van. Tide: threat identification using active dns measurements. In: ACM. **Proceedings of the SIGCOMM Posters and Demos**. [S.l.], 2017. p. 65–67.

STEWART, W. **Living Internet**. 2019. <https://www.livinginternet.com/>. Accessed 13 Jun. 2018.

VIXIE, P.; SNEERINGER, G.; SCHLEIFER, M. **Events of 21-Oct-2002**. 2002. <http://c.root-servers.org/october21.txt>. Accessed 13 Jun. 2018.

WEINBERG M., W. D. **Review and analysis of attack traffic against A-root and J-root on November 30 and December 1, 2015**. 2016. <https://indico.dns-oarc.net/event/22/session/4/contribution/7>. Accessed 25 Aug. 2018.

APPENDIX A — ACCEPTED PAPER – AINA 2020

The Internet Domain Naming System (DNS) is one of the pillars for the Internet and has been the subject of various Distributed Denial-of-Service (DDoS) attacks over the years. As a countermeasure, the DNS infrastructure has been engineered with a series of replication measures, such as relying on multiple authoritative name servers and using IP anycast. Even though these measures have been in place, we have seen that, when servers rely on third-party DNS providers for reliable services, there may be certain levels of infrastructure centralization. In this case, an attack against a DNS target might affect other authoritative DNS servers sharing part of the infrastructure with the intended victim. However, measuring such levels of infrastructure sharing is a daunting task, given that researchers typically do not have access to DNS provider internals. In this paper, we introduce a methodology and associated tool `dnstracker` that allows measuring, to various degrees, the level of both concentration and shared infrastructure using active DNS measurements. As a case study, we analyze the authoritative name servers of all domains of the Alexa Top 1 Million most visited websites. Our results show that, in some cases, up to 12.000 authoritative name servers share the same underlying infrastructure of a third-party DNS provider. As such, in the event of an attack, those authoritative DNS servers have increased the probability of suffering from collateral damage.

- **Title:** `dnstracker`: Measuring Centralization of DNS Infrastructure in the Wild
- **Conference:** International Conference on Advanced Information Networking and Applications (AINA)
- **Type:** Main Track (Full Paper)
- **Qualis:** A2
- **Held at:** Caserta, Italy

`dnstracker`: Measuring Centralization of DNS Infrastructure in the Wild

Luciano Zembruzki, Arthur Selle Jacobs, Gustavo Spier Landtreter, Giovane C. M. Moura and Lisandro Zambenedetti Granville

[1]Institute of Informatics – Federal University of Rio Grande do Sul

Abstract. The Internet Domain Naming System (DNS) is one of the pillars for the Internet and has been the subject of various Distributed Denial-of-Service (DDoS) attacks over the years. As a countermeasure, the DNS infrastructure has been engineered with a series of replication measures, such as relying on multiple authoritative name servers and using IP anycast. Even though these measures have been in place, we have seen that, when servers rely on third-party DNS providers for reliable services, there may be certain levels of infrastructure centralization. In this case, an attack against a DNS target might affect other authoritative DNS servers sharing part of the infrastructure with the intended victim. However, measuring such levels of infrastructure sharing is a daunting task, given that researchers typically do not have access to DNS provider internals. In this paper, we introduce a methodology and associated tool `dnstracker` that allows measuring, to various degrees, the level of both concentration and shared infrastructure using active DNS measurements. As a case study, we analyze the authoritative name servers of all domains of the Alexa Top 1 Million most visited websites. Our results show that, in some cases, up to 12.000 authoritative name servers share the same underlying infrastructure of a third-party DNS provider. As such, in the event of an attack, those authoritative DNS servers have increased the probability of suffering from collateral damage.

Keywords: Domain Name System, Measurements, Centralization

1 Introduction

The Internet Domain Naming System (DNS) provides a globally hierarchical naming space on the Internet that enables the mapping of hosts, networks, and services to IP addresses [1]. As such, DNS is one of the core services of the Internet. To resolve a domain name (*e.g.*, `ufrgs.br`), first, a client sends a DNS query to its *DNS recursive resolver* (resolver hereafter), which is a DNS server that, on behalf of the client, can resolve the domain name. If the resolver does not have a DNS record in a cache, it will query the DNS hierarchy for a response. Resolvers are responsible for sending queries to *DNS authoritative nameservers* (authoritative DNS server hereafter), which are the servers responsible for providing answers to resolvers about the fetched domain. These authoritative DNS

servers are divided into zones and only know the content of a DNS zone from local knowledge, and thus can answer queries about those zones [2].

The authoritative DNS servers have been frequent victims of Distributed Denial-of-Service (DDoS) attacks. The Root Zone, which is authoritative for the Root (`.`) DNS zone, has been targeted various times in the last decade [3–7], and even DNS providers have been victims of attacks [8], disrupting many of its domains [9]. DNS has been engineered with *layers of replication* to curb such attacks: first, a domain name may use multiple authoritative name servers. Second, each authoritative name server may employ IP anycast [10], which allows the same IP addresses to be replicated and announced from various locations, referred to as anycast sites. Third, each site, in turn, may locally use load balancers to distribute queries among multiple servers [7], increasing reliability even further.

Even though these measures are broadly employed, when domain names share the same DNS provider, they may be (unknowingly or not) sharing different levels of *infrastructure*, such as pipes, servers, and data centers. As many companies do not run their DNS infrastructure, instead of outsourcing to third-party DNS providers, identifying possible infrastructure sharing among many distinct domains, becomes a challenging endeavor. This infrastructure sharing may become a problem when a large enough DDoS attack takes place: if parts of the shared infrastructure become overwhelmed, all DNS zones under the service may experience problems too. As a consequence, many domains under zones may become unreachable. The Dyn attack [8] exemplifies the *collateral damage* when authoritative servers hosting multiple DNS zones are under attack.

The Internet DNS has been analyzed and studied by multiple authors [11–13], yet few works focused on measuring the levels of the shared DNS infrastructure. Besides, measuring such levels of infrastructure sharing is a daunting task, given that researchers typically do not have access to DNS provider internals. As such, researchers have to resort to active measurements that allow to estimate, at the IP level, a certain degree of shared infrastructure, or analyze historical DNS datasets. This study has been done previously in some studies [14–16] that analyzed different aspects of the DNS ecosystem, such as the robustness and centralization of Top-Level Domains (TLD) [16] [14] and Root servers [15]. Despite shedding some light on infrastructure sharing at the TLD level by providing evidence that network-level infrastructure sharing is becoming more frequent over time. Those studies do not inspect DNS centralization on an Autonomous System (AS) level, derived from relying solely on third-party DNS providers. Also, the authors did not provide any analysis of the shared infrastructure of authoritative DNS servers for Fully Qualified Domain Names (FQDNs).

Given this scenario, we introduce in this paper a methodology that allows measuring, to various degrees, the level of centralization of authoritative DNS servers using active DNS measurements. We focus our work on analyzing a possible centralization in authoritative DNS servers in the wild, for FQDNs. Also, we developed `dnstracker`, an opensource tool that implements our proposed methodology and provides a consolidated view of our findings. As a case study,

we use `dnstracker` to analyze all domains of Alexa Top 1 Million [17] websites. We show that, in some cases, up to 12,000 authoritative DNS servers of the most visited websites share the same infrastructure of a DNS provider, and as such, could suffer from collateral damage in the event of an attack.

The remainder of this paper is organized as follows. In Section 2, we discuss the Related Work, reviewing previous efforts that analyzed DNS and its infrastructure. In Section 3, we describe the `dnstracker` methodology used to measure the DNS centralization and discuss its efficiency. In Section 4, we present our results. Finally, in Section 5, we conclude this work and discuss future directions.

2 Related Work

Moura *et al.* [15] analyzed the DDoS event suffered by the DNS Root servers in 2015. Between Nov. 30th to Dec. 1st, 2015, many of the Root DNS Letter Servers had an unusually high rate of a specific request, with a traffic rate a hundred times larger than the normal load. The authors highlighted that, even though these episodes did not target specific end-services, there was evidence of Internet services suffering from *collateral damage* because of sharing the DNS provider infrastructure with the DDoS target. In the 2015 attack, some `.nl` TLD servers were taken down as a side effect from the attack to the root DNS server. Even though that investigation provided a diagnosis of the events and highlighted some shreds of evidence of shared infrastructure, it did not investigate in depth the possible level of centralization in the DNS ecosystem.

Bates *et al.* [16] proposed a solution to measure how far the global DNS has preserved its distributed resilience, given the rise of cloud-based hosting and infrastructure. In their work, the authors analyzed the trends in concentration and diversification of the DNS infrastructure over time, where they sampled the 1,000 main US domains in the TLDs `.com`, `.net`, and `.org` according to Alexa Top Sites [17]. The authors also pointed out that their analysis focused on the traditional domains `.com`, `.net`, and `.org` because they are among the oldest TLDs, thus representing a broad spectrum of the current Internet. However, the authors recognize that their results might change if other TLDs, such as `.ru` and `.cn`, were taken into account. Despite providing some insight into the robustness of DNS, the work did not consider the possible concentration of authoritative DNS servers, which is a crucial point in infrastructure reliability. That, in turn, is covered by our work.

Allman *et al.* [14] carried out a study to observe the robustness of the DNS ecosystem. Their analysis was focused on Second-Level Domains (SLDs) (*e.g.*, , `icir.org`). In that study, the authors used two sets of zone files for the `.com`, `.net`, and `.org` TLDs. That data was collected over nine years. They performed an analysis of DNS infrastructure sharing. Initially, it was noted that 91% to 93% of the observed SLDs share, at least, one name server (by IP) with at worst one another SLD. In an approach based on individual SLDs, the authors observed that half of the SLDs share exactly one set of authoritative DNS servers with

at the very least 163 other SLDs. Also, it was discovered that the largest group contains 9,000 SLDs that share the same set of authoritative DNS servers. In further analysis, by looking for shared infrastructure over IP blocks instead of single IPs, the authors found an even greater level of concentration. Besides, the authors point out that such network-level infrastructure sharing is becoming more common over time. Finally, they analyze the data to determine whether shared infrastructure occurs more frequently in domains with a higher or lower ranking. Their study, however, did not point to any general result or specific trends.

Considering the research carried out so far in the scientific community, there is strong evidence that suggests some level of DNS centralization. However, none of the works in the state-of-the-art has considered the centralization of authoritative DNS servers for FQDNs. Besides, it is also essential to have in mind not only the last hop but the hop before the last one, which is part of the contribution of our work. In Section 3, we describe our methodology to identify and quantify the centralization of the global DNS infrastructure.

3 `dnstracker`

The outsourcing trend of DNS providers poses several challenges to identifying possible infrastructure sharing. Also, the collateral damage cannot be directly assessed by a simple analysis of the IP addresses of different authoritative DNS servers. The different servers - each under their own IP address block - could be hosted behind the same service provider's infrastructure. In addition, due to the commercial nature of DNS providers, the data required to analyze such aggregation is rarely disclosed. An indication of this problem may be the presence of a common node (or single point of failure) in the routing path for a specific set of remote servers. For instance, if we get the IP address of `a.dns.nl`, one of the authoritative servers for the `.nl` zone, and examine its AS, we will find it belongs to a DNS provider. In fact, this authoritative DNS server is run by NetNod in Sweden. Hence, if other authoritative DNS servers are hosted on the same Netnod infrastructure, they start sharing the collateral damage in a potential DDoS attack. Below, we describe our proposed methodology for measuring DNS centralization, as well as the system we developed to implement it.

3.1 Measuring centralization

We propose an approach that identifies common points in routing paths. Initially, for a given FQDN, we use a `dig` Linux command to uncover its authoritative DNS server. An FQDN normally has from two to four distinct authoritative DNS servers, but many domains share the same authoritative DNS server. Then, server uncovered with the dig command, we execute a custom `traceroute` command from a given vantage point. The command provides information about the addresses of each hop in the route from the vantage point to the domain's

authoritative DNS server. Whenever a set of distinct servers, owned by different websites, are hosted behind the same provider's infrastructure, requests to these servers will share a common point - the network hop just before reaching it's final destination, referred to as *Hop-Before-The-Last* (HBTL). If two different requests are served through a path whose HBTL are in the same AS, they are likely hosted by the same DNS provider, thus sharing the same infrastructure to some extent, as illustrated by Figure 1.

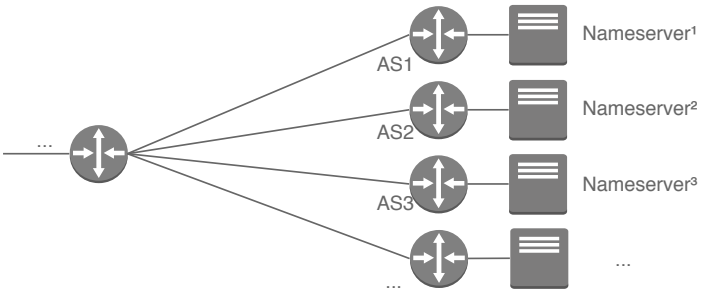


Fig. 1. Name Servers whose HBTL share the same AS infrastructure

From each ICMP Echo Reply obtained through *traceroute*, we extract the IPv4 address of each hop in the path to the authoritative DNS server. However, in our approach we only store the relevant data of the last hop and the HBTL, as these are the most likely points of aggregation of infrastructure in the DNS ecosystem. For each IPv4 address, we use a publicly available BGP table [18] to obtain the corresponding AS of the hop, as well as the owner of the hop (*i.e.*, what company is responsible for running the infrastructure). We repeat this step for both the last hop and the HBTL. Listing 1.1, Listing 1.2 and Listing 1.3 present a step by step exemple of our methodology, using `bash` commands as examples of our proposed approach. For this example domain, we can see that authoritative DNS servers are hosted with in-house infrastructure of UFRGS, since both the ASes of the last hop and the HBTL are the same.

```
host $>dig ufrgs.br
;; AUTHORITY SECTION:
ufrgs.br 3600 IN NS ns1.ufrgs.br
ufrgs.br 3600 IN NS ns2.ufrgs.br
```

Listing 1.1. Uncovering domain authoritative DNS servers.

```
host $>traceroute ns1.ufrgs.br
...
19 ge-0-0-2-0.arn1-rndfw1.as1140.nl (94.198.152.11)
20 proteus.ufrgs.br (94.198.159.3)
```

Listing 1.2. Uncovering IPv4 addresses of last hop and HBTL

```
host $>ip2asn 94.198.152.11
AS Number: 1140, AS Description: UFRGS
host $>ip2asn 94.198.159.3
AS Number: 1140, AS Description: UFRGS
```

Listing 1.3. Mapping ASes of last hop and HBTL

Finally, having received the responses of all hops until the targeted authoritative DNS servers, and mapping the corresponding ASes of each hop, we store this information in our database for further analysis. When executedly repeatedly, we are able to consolidate the millions of entries in the database to identify a possible aggregation of infrastructure, in many different levels, as well as analyze the changes in the DNS ecosystem over time.

3.2 System Architecture

To support our methodology, the `dnstracker` tool was developed to collect DNS-related information and to expose the level of centralization of the DNS infrastructure. The source code for `dnstracker` is publicly available at GitHub[1]. Figure 2 presents the architecture of `dnstracker`.

`dnstracker` Agent. On the left side of the Figure 2, a group of `dnstracker` Agents retrieve information, using `traceroute`, from target authoritative DNS servers (1). The target servers are obtained from the list of the world's most popular websites, provided by Alexa Top 1 Million domain list open repository [17], accessed in January 2018 (hosted as a local conceptual database inside each agent). The agent applies our collection methodology for each domain in the list. After discovering information from all authoritative DNS servers, the `dnstracker` Agent exports the created datasets to the `dnstracker` Server (2) using a REST API [19]. It is also important to mention that the traditional Linux implementation of `traceroute` does not support parallelism and, hence, it is not fit for actively crawling through large numbers of domains. We then implemented a custom version of `traceroute` with parallelism support, using Java, running inside each agent.

`dnstracker` Server. After exporting the collected data from the `dnstracker` agent, the `dnstracker` Server processes datasets to create appropriate visualization and exposes them for the system's users via HTTP (3). We used the Spring Boot v2.0.4 framework to prototype this Web user interface[2].

One of the benefits of `dnstracker` is that the tool automates several otherwise labor-intensive tasks. That includes tracing the route to all authoritative DNS servers in the database every month, identifying which ASes the last hop and HBTL belong to, as well consolidating the data to present possible aggregation. By using `dnstracker`, the user can observe various aspects of the DNS

[1] https://github.com/ComputerNetworks-UFRGS/dnstracker
[2] http://dnstracker.inf.ufrgs.br

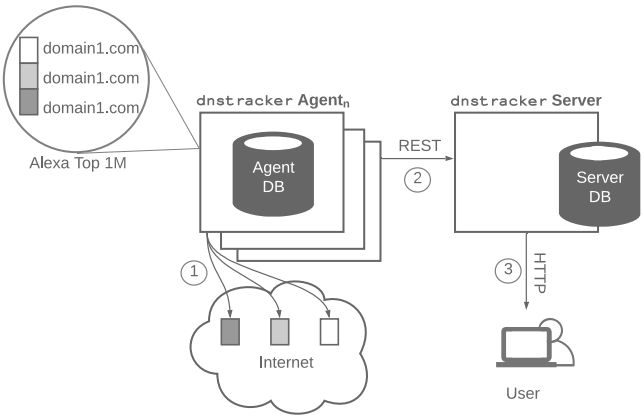


Fig. 2. `dnstracker` architecture

centralization. In the next Section, and through the use of `dnstracker`, we present the current picture of the Internet's DNS centralization.

4 Results

The application `dnstracker` that implements our methodology was deployed on two instances of Amazon EC2 in Sao Paulo, Brazil. One of this instances was used as a `dnstracker` Agent and the other as a `dnstracker` Server. This instances have the same configuration with a single core CPU and RAM memory of 0.5GB. We executed our measurements several times a month, from January 2018 to May 2018, resulting in a dataset of millions of *traceroute* entries.

In this section, we present the results obtained throught our proposed methodology. Having obtained the dataset through `dnstracker`, We focus on three facets to identify possible infrastructure centralization as well identify possible risk of collateral damage. First, we evaluate the concentration of authoritative servers per last hop AS. Second, we measure the concentration of authoritative server ASes per HBTL AS. Third, we determine the total number of authoritative servers that shared the same HBTL. These three aspects enable us to measure the amount of authoritative DNS servers that share AS infrastructure with other authoritative DNS servers, at both last hop level and HBTL level. Finally, we analyze whether, among the top DNS providers, there is any growth trend in these aggregations over the measurement period. First, we describe our datasets. After that, we present our results.

4.1 Datasets

Table 1 presents a summary of the data collected throughout the five months of observations. Over the measurement period, the number of observed distinct authoritative DNS servers, ASes, and HBTL ASes remained stable. In addition,

the number of distinct authoritative DNS servers is much smaller than the number of investigated domains, since many domains share the same authoritative authoritative servers. In fact, among the data we collected, one single server was authoritative for over 10,000 domains, belonging to a big DNS provider: DNSPod. This does not definitely mean that the servers would lead to problems, since there may be multiple design measures in place to increase its the fault tolerance against DDoS attacks, but it also reveals the existence of actual shared infrastructure. In our samples, 136,421 out of the traced authoritative DNS servers had ASes in their routes that explicitly discarded ICMP echo requests, which hindered obtaining information about the HBTL of such server. Because of that, in the analysis of HBTL aggregation such authoritative DNS servers were disregarded; circumventing this observation is a subject of future work in our research.

	DNS data			`traceroute` data	
Month	NS rec.	IPv4 (NS)	Last Hop ASes	HBTL IPv4	HBTL ASes
Jan - May	283,983	208,543	18,400	40,157	7,742

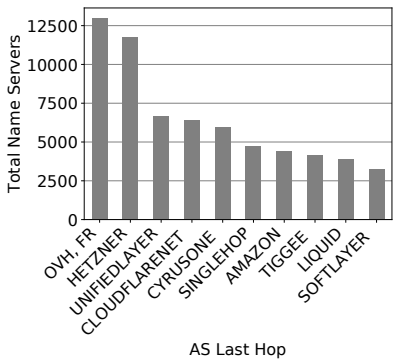
Table 1. Datasets generated by `dnstracker` for 2018 monthly measurements for Alexa 1 million domain names.

Bellow, in the Section 4.2 we show our results about the concentration of authoritative servers per last hop AS.

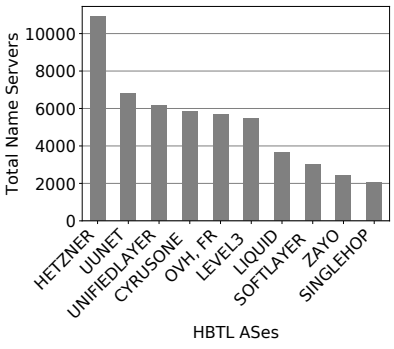
4.2 Aggregation of Authoritative DNS Server

First, we analyze the concentration of distinct authoritative DNS servers by last hop ASes. As shown in Figure 3(a) "OVH, FR" is the provider with the largest number of authoritative DNS servers in its infrastructure, aggregating more than 12,000 distinct authoritative DNS servers, each with multiple hosted domains. This means that, in case of a successful attack to this AS, over 77,419 websites would be unreachable, since clients would not be able to resolve FQDNs. Then, we can observe that the provider "HETZNER-AS, DE", holds 11,000 of the total authoritative servers, which represents 30,947 distinct websites hosted, followed by "UNIFIEDLAYER-AS-1 - Unified Layer, US", which hosts 6,000 authoritative servers behind his infrastructure representing 5,825 distinct websites. By analyzing the collected data, we can see that, hosting an authoritative DNS server in these 3 providers presents a higher risk of suffering from collateral damage. These providers concetrate a large portion of the analyzed domains, and hence would likely be more target by attacks. In the other providers observed in Figure 3(a), we can see a margin ranging from 6,000 to 3,000 of the total authoritative DNS servers in each of the providers. These observations match with the results obtained in previous studies [14] by observing the IP blocks of the authoritative DNS servers. As indicated by previous work [16, 14], we also reinforce

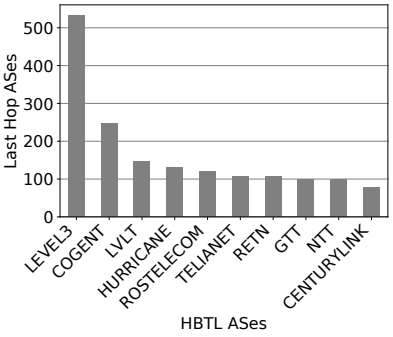
the presence of centralization in DNS services. We understand that large DNS provider such as these often offer multiple levels of redundancy to curb possible attacks. However, it is worth pointing out that DDoS attacks are becoming increasingly sophisticated, including o attack to the DynDNS infrastructure [8], so this should be a point of attention.



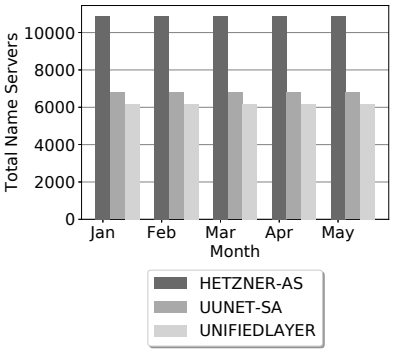
(a) Authoritative DNS server Aggregation by Last Hop AS



(b) Authoritative DNS server Aggregation by HBTL



(c) AS Aggregation by HBTL



(d) Authoritative DNS server aggregation by HTBL ASes over time

Fig. 3. Authoritative DNS server Aggregation

4.3 Aggregation of Authoritative DNS Servers by HBTL ASes

In addition to analyzing the aggregation of authoritative DNS servers in the last hop, we inspect the amount of authoritative DNS servers that share the same HBTL, since it can possibly be a single point of failure. Figure 3(b) present the top 10 HBTL ASes that aggregated more authoritative DNS servers.

The provider identified by "HETZNER-AS, DE" in Figure 3(b), shows that almost 11,000 of the total authoritative DNS servers share this same hop as its HBTL. We mention that HBTL may change depending on the vantage point. Other vantage points will be analyzed in future work. The "UUNET-SA - MCI Communications Services, Inc.", is shared by almost 7,000 authoritative server as well. These numbers suggest the presence centralization in the DNS infrastructure itself, not only at the last hop, as mentioned by previous studies [16][14], but also in HBTL as well. In addition, it is important to highlight once more that of these authoritative DNS servers resolve more then 100,000 of domains. Hence, if a HBTL was to be taken down, hundred of thousands of domains would become unreachable as collateral damage.

4.4 Aggregation of Last Hop ASes by HBTL ASes

Up to here, we focused on analyzing the concentration of authoritative DNS servers in each hop. However, when looking to third-party provider ASes, other services, such as storage, database, emails may be affected in addition to the hosted authoritative DNS servers. Hence, we also study the number of distinct ASes that share the same HBTL AS, aiming to identify points of shared network infrastructure that might represent possibility of collateral damage to authoritative DNS servers, *i.e.*, a totally unrelated service might be targeted by an attack and still affect the DNS ecosystem because of shared infrastructure.

As we can see in Figure 3(c), in this assessment, the most noteworthy aggregation of last hop ASes occurs in the "LEVEL3 - Level 3 Communications, Inc., US" HTBL AS. Level 3 is one of the top infrastructure providers in the world, so this is a natural result. However, the number of last hop ASes that share its infrastructure is large, amounting to over 500 different ASes. The second largest HBTL aggregation, provider "COGENT-174 - Cogent Communications, US", has less than half of Level 3's amount, with 200 AS behind it. Although the concentration of ASes behind a single hop has probably more to do with delivery structure than surely on DNS services, such a concentration is increases the chance of problems for a larger amount of service if targeted by a large-scale attack.

4.5 Summary

The analysis we provided in the previous subsection showed a considerable amount of infrastructure sharing, in many different levels of the DNS ecosystem. In particular, the level of concentration of authoritative DNS servers by HBTL is worth highlighting, as most DNS operators are not aware of such concentration when contracting hosting services. Looking solely at the last hop ASes for concentration, on the other side, may be misleading because many companies (*e.g.*, `facebook.com`) may advertise their own ASes for their authoritative servers but still rely on the infrastructure of a third-party provider. In such cases, the possibility of collateral damage is still present, but undetected so far.

Lastly, as we evaluate DNS aggregation over 5 months, we must look at the difference in HBTL aggregation of authoritative servers during this period. By doing so, one may be able to identify if a trend of centralizing the DNS infrastructure in fact exists, at least in such a period. Figure 3(d) presents the aggregation level of the top 3 HBTL ASes, over each month we traced the authoritative servers. By observing the temporal graph, the centralization of authoritative servers of the Alexa Top 1 Million remained stable in the period. This is consistent with the general assumption that the DNS infrastructure is stable and robust. Also, that can be justified by the fact that the observed providers offer reliability and there is no need for frequent changes in hosting them. However, this does not mean that there is no centralization trend considering a larger time window.

5 Conclusions and Future Work

In this paper, we presented `dnstracker`, a tool to measure the centralization and shared infrastructure of Internet's DNS using active DNS measurements. `dnstracker` implements our proposed methodology that relies on `traceroute` to trace and get the informations about the last two hops of a authoritative DNS servers. We focus our work on analyzing centralization in the DNS infrastructure in the wild, for FQDNs. As a case study, we used `dnstracker` to analyze all domains of the Alexa Top 1 Million [17] websites. The analysis showed a considerable amount of infrastructure sharing, in many different levels of the DNS ecosystem. We show that, in some cases, up to 12,000 authoritative DNS servers of the most visited websites share the same infrastructure of big DNS providers, and thus could suffer from collateral damage in the event of an attack. In addition, we analyzed our measurements collected during 5 months to try to identify a centralization trend in the DNS global infrastructure. However, no trend was identified in the period of time we collected our data.

The future directions of our research include observing DNS centralization from different vantage points in the Internet; we want to understand how influential a vantage point is in our observation methodology. We also want to exploit the Ripe Atlas infrastructure to carry out our analysis. Finally, at a more theoretical perspective, we are working on a centralization metric that will help network operators find the more appropriate hosts for their DNS needs.

References

1. P. Mockapetris, "Domain names - concepts and facilities," Internet Requests for Comments (RFC), Internet Engineering Task Force, STD 13, November 1987.
2. R. Elz, R. Bush, S. Bradner, and M. Patton, "Selection and Operation of Secondary DNS Servers," RFC 2182 (Best Current Practice), RFC Editor, Fremont, CA, USA, pp. 1–11, Jul. 1997. [Online]. Available: https://www.rfc-editor.org/rfc/rfc2182.txt
3. Paul Vixie and Gerry Sneeringer and Mark Schleifer, "Events of 21-oct-2002," Oct. 2002, http://c.root-servers.org/october21.txt.

4. Root Server Operators, "Events of 2015-11-30," Nov. 2015, http://root-servers.org/news/events-of-20151130.txt.
5. Root Server Operators, "Events of 2016-06-25," Root Server Operators, Tech. Rep., June 29 2016. [Online]. Available: http://www.root-servers.org/news/events-of-20160625.txt
6. Weinberg, M., Wessels, D., "Review and analysis of attack traffic against A-root and J-root on November 30 and December 1, 2015," In: DNS OARC 24 – Buenos Aires, Argentina. https://indico.dns-oarc.net/event/22/session/4/contribution/7, April 2016.
7. G. C. M. Moura, R. de O. Schmidt, J. Heidemann, W. B. de Vries, M. Müller, L. Wei, and C. Hesselman, "Anycast vs. DDoS: Evaluating the November 2015 root DNS event," in *Proceedings of the ACM Internet Measurement Conference*, Nov. 2016. [Online]. Available:
8. S. Hilton, "Dyn analysis summary of Friday October 21 attack," Dyn blog https://dyn.com/blog/dyn-analysis-summary-of-friday-october-21-attack/, Oct. 2016.
9. N. Perlroth, "Hackers used new weapons to disrupt major websites across U.S." *New York Times*, p. A1, Oct. 22 2016. [Online]. Available: http://www.nytimes.com/2016/10/22/business/internet-problems-attack.html
10. D. McPherson, D. Oran, D. Thaler, and E. Osterweil, "Architectural Considerations of IP Anycast," RFC 7094 (Informational), RFC Editor, Fremont, CA, USA, pp. 1–22, Jan. 2014. [Online]. Available: https://www.rfc-editor.org/rfc/rfc7094.txt
11. L. Bilge, S. Sen, D. Balzarotti, E. Kirda, and C. Kruegel, "Exposure: A passive dns analysis service to detect and report malicious domains," *ACM Transactions on Information and System Security (TISSEC)*, vol. 16, no. 4, p. 14, 2014.
12. R. Perdisci, I. Corona, and G. Giacinto, "Early detection of malicious flux networks via large-scale passive dns traffic analysis," *IEEE Transactions on Dependable and Secure Computing*, vol. 9, no. 5, pp. 714–726, 2012.
13. A. A. Mugali, A. W. Simpson, S. K. Walker *et al.*, "System and method for detecting dns traffic anomalies," Oct. 27 2015, uS Patent 9,172,716.
14. M. Allman, "Comments on DNS Robustness," in *ACM Internet Measurement Conference*, Nov. 2018, to appear.
15. "Anycast vs. DDoS: Evaluating the November 2015 Root DNS Event," in *Proceedings of the 2016 ACM on Internet Measurement Conference - IMC '16*, no. November 2015, 2016. [Online]. Available: http://dl.acm.org/citation.cfm?doid=2987443.2987446
16. S. Bates, J. Bowers, S. Greenstein, J. Weinstock, and J. Zittrain, "Evidence of decreasing internet entropy: The lack of redundancy in dns resolution by major websites and services," National Bureau of Economic Research, Tech. Rep., 2018.
17. Alexa, "Alexa Top 1 Million," Jan. 2018, http://s3.amazonaws.com/alexa-static/top-1m.csv.zip.
18. RIPE Network Coordination Centre, "RIPE Atlas." [Online]. Available: https://atlas.ripe.net/
19. R. T. Fielding, "Architectural styles and the design of network-based software architectures," Ph.D. dissertation, 2000, university of California, Irvine.

APPENDIX A — RESUMO

O Sistema de Nomes de Domínios (*Domain Name System* - DNS) é um sistema hierárquico que permite o mapeamento de hosts, redes e serviços para endereços IP (MOCKAPETRIS, 1987a). Como tal, o DNS é um dos principais serviços da Internet. Para resolver um nome de domínio (por exemplo, `example.nl`, um cliente geralmente envia uma consulta DNS para seu *resolvedor recursivo* (resolvedor daqui em diante), que é um servidor DNS que, em nome do cliente , pode resolver o nome do domínio. Os resolvedores são responsáveis por enviar consultas para *servidores de nomes autoritativos*, que são servidores que conhecem o conteúdo de uma zona DNS a partir do conhecimento local e, portanto, podem responder a consultas sobre essas zonas (ELZ et al., 1997). Por exemplo, um cliente pode se conectar a um resolvedor de DNS público (*i.e*, `1.1.1.1` (1.1.1.1, 2018)) solicitando o IP de `example.nl`. O resolvedor irá, em nome do usuário, enviar consultas para os servidores oficiais de `example.nl`, que são `ex1.sidnlabs.nl` e `ex2.sidnlabs.nl`, e retornar o endereço IP desejado para o cliente.

Servidores DNS autoritativos têm sido vítimas frequentes de ataques de negação de serviço distribuido (DDoS - Distributed Denial-of-Service). Por exemplo, o sistema DNS Root, que é autoritativo para a zona DNS Root `(.)`, foi alvo e ameaçado várias vezes na última década (VIXIE; SNEERINGER; SCHLEIFER, 2002) (OPERATORS, 2015) (OPERATORS, 2016) (WEINBERG M., 2016) (MOURA et al., 2016) (SENGUPTA, 2012). Outros servidores DNS autoritativos também foram atacados. Em 2016, a Dyn, um dos maiores provedores de DNS, foi vítima de um ataque de 1.3Tb/s da botnet IoT Mirai (HILTON, 2016), resultando em interrupções de vários sites que usam a Dyn, incluindo Twitter, Netflix e o New York Times (PERLROTH, 2016).

Embora medidas tenham sido projetadas para minimizar problemas, vimos que, quando nomes de domínio empregam o mesmo provedor de DNS, eles podem estar (sem saber ou não) compartilhando diferentes níveis de infraestrutura, como servidores e datacenters. Esse compartilhamento de infraestrutura pode se tornar um problema quando ocorre um ataque DDoS grande o suficiente: se partes da infraestrutura compartilhada ficarem sobrecarregadas, todas as zonas DNS do serviço também poderão ter problemas. Como consequência, muitos domínios sob zonas podem se tornar inacessíveis. O ataque Dyn exemplifica o risco de *dano colateral* quando servidores autoritativos que hospedam várias zonas DNS estão sob ataque.

A medição de tais níveis de compartilhamento de infraestrutura é uma tarefa difícil, uma vez que os pesquisadores normalmente não têm acesso aos internos do provedor de DNS. Como tal, os pesquisadores precisam recorrer a medições ativas, que permitem estimar, no nível do IP, um certo grau de infraestrutura compartilhada, ou analisar conjuntos de dados históricos do DNS. Isso foi feito anteriormente por Allman (ALLMAN, 2018), onde o autor analisou os arquivos de zona DNS para `.org`, `.com` e `.net` cobrindo um período de nove anos. O autor analisou a infraestrutura compartilhada do DNS em termos de blocos de endereços IP com prefixo de 32 bits. de endereços IP. Entre as descobertas, Allman mostra que, por exemplo, um único servidor autoritativo era responsável por 9.000 zonas DNS. No entanto, o estudo se concentrou em explorar o compartilhamento de infraestrutura em um nível de IP por hosts individuais de blocos de endereços. Os autores ressaltam que o compartilhamento de infraestrutura em nível de rede está se tornando mais comum ao longo do tempo, mas não inspecionaram o compartilhamento de infraestrutura no nível do Sistema Autônomo (AS), derivado da dependência exclusiva de provedores de DNS de terceiros. Além disso, os autores não forneceram nenhuma análise da infraestrutura compartilhada de servidores de nomes oficiais para FQDN (Fully Qualified Domain).

Diante desse cenário, apresentamos neste trabalho uma solução que permite medir, em vários graus, o nível de concentração e a infraestrutura compartilhada usando medições de DNS ativas. Focamos nosso trabalho na análise de uma possível centralização em ASes na internet, para FQDNs. Além disso, como estudo de caso, usamos o `dnstracker` para analisar todos os domínios da lista Alexa Top 1 Million.